Naval Research Laboratory

Washington, DC 20375-5320

NRL/MR/5320--07-9029

A Generalized FFT for Many Simultaneous Receive Beams

JEFFREY O. COLEMAN

Advanced Radar Techniques Branch
Radar Division

June 29, 2007

Approved for public release; distribution is unlimited.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing this collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. **PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.**

1. REPORT DATE *(DD-MM-YYYY)*
29-06-2007

2. REPORT TYPE
Memorandum Report

3. DATES COVERED *(From - To)*

4. TITLE AND SUBTITLE

A Generalized FFT for Many Simultaneous Receive Beams

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER
61153N

6. AUTHOR(S)

Jeffrey O. Coleman

5d. PROJECT NUMBER

5e. TASK NUMBER
EW021-05-43

5f. WORK UNIT NUMBER
8574

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Naval Research Laboratory
4555 Overlook Avenue, SW
Washington, DC 20375-5320

8. PERFORMING ORGANIZATION REPORT NUMBER

NRL/MR/5320--07-9029

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSOR / MONITOR'S ACRONYM(S)

11. SPONSOR / MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

It is well known that when the identical elements of a planar receive array are laid out in horizontal rows and vertical columns, a fast Fourier transform or FFT can be used to efficiently realize simultaneous beams laid out in rows and columns in the direction cosines associated with the azimuth and elevation directions. Here a more general formulation and an associated design discipline is developed. Identical elements are laid out on an arbitrary planar lattice — it could be square, rectangular, diamond, or triangular and might display tremendous symmetry or very little — and the beams in direction-cosine space are laid out on an arbitrary superlattice of the dual of the element-layout lattice. The generality of these two arbitrary lattices can yield significant cost reductions for large, many-beam arrays and arises from, first, formulating the desired beam outputs using a discrete Fourier transform or DFT generalized to use an integer-matrix size parameter, and second, efficiently realizing the required real-time computations with the generalized FFT based on a matrix factorization of that size parameter that is developed here. This generalized FFT includes as special cases the usual 1D and 2D FFT's in radix-2 and mixed-radix forms but offers many more possibilities as well. The approach cannot outperform but does match, when the matrix size parameter factors well, the N log N computational efficiency of the usual FFT. Examples illustrate a design discipline for the two lattices that involves jointly determining element spacing, steering range and beam-layout geometry, grating-lobe behavior, and FFT factorability and therefore computational efficiency.

15. SUBJECT TERMS

Array radar	FFT beam steering
Digital arrays	Generalized FFT

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
					Jeffrey O. Coleman
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**	UL	43	**19b. TELEPHONE NUMBER** *(include area code)*
Unclassified	Unclassified	Unclassified			(202) 404-8843

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std. Z39.18

Contents

1 Introduction **1**

2 Multi-beam steering with a matrix-size DFT **3**
2.1 Remaindered division of an integer vector by a square integer matrix 4
2.2 The 2D lattice array from first principles 10
2.3 Many Beams from the DFT 14

3 System-design examples **17**
3.1 The Classic $\lambda/2$ square-grid design 17
3.2 The Classic $\lambda/\sqrt{3}$ triangular-grid design 24
3.3 The general case: custom replication and steering lattices 27

4 A 2D FFT with a matrix size parameter **31**
4.1 General derivation 31
4.2 Familiar special cases 34
4.3 Custom FFT design 35

5 Concluding Remarks **37**

References **39**

A Generalized FFT for Many Simultaneous Receive Beams

Dr. Jeffrey O. Coleman*

Naval Research Laboratory
Radar Division
Advanced Radar Techniques Branch
Code 5328, Signal Processing Theory and Methods Section

Abstract

It is well known that when the identical elements of a planar receive array are laid out in horizontal rows and vertical columns, a fast Fourier transform or FFT can be used to efficiently realize simultaneous beams laid out in rows and columns in the direction cosines associated with the azimuth and elevation directions. Here a more general formulation and an associated design discipline is developed. Identical elements are laid out on an arbitrary planar lattice—it could be square, rectangular, diamond, or triangular and might display tremendous symmetry or very little—and the beams in direction-cosine space are laid out on an arbitrary superlattice of the dual of the element-layout lattice.

The generality of these two arbitrary lattices can yield significant cost reductions for large, many-beam arrays and arises from, first, formulating the desired beam outputs using a discrete Fourier transform or DFT generalized to use an integer-matrix size parameter and, second, efficiently realizing the required real-time computations with the generalized FFT based on a matrix factorization of that size parameter that is developed here. This generalized FFT includes as special cases the usual 1D and 2D FFT's in radix-2 and mixed-radix forms but offers many more possibilities as well. The approach cannot outperform but does match, when the matrix size parameter factors well, the $N \log N$ computational efficiency of the usual FFT.

Examples illustrate a design discipline for the two lattices that involves jointly determining element spacing, steering range and beam-layout geometry, grating-lobe behavior, and FFT factorability and therefore computational efficiency.

*This work was supported by the Defense Advanced Projects Research Agency (DARPA) Strategic Technologies Office (STO) and by the Naval Research Laboratory base program.

1 Introduction

Let us orient ourselves to the problem by recalling a few basics. The *discrete Fourier transform* or DFT $X_k = \sum_{n=0}^{N-1} x_n \, \mathrm{e}^{-\mathrm{j}2\pi kn/N}$ transforms a function x_n on finite time domain $0, \ldots, N-1$, the set of all possible remainders modulo N, to a function X_k on a finite frequency domain comprising the same possible remainders.

The *fast Fourier transform* or FFT is any of a wide variety of efficient recursive computational approaches to computing the DFT. The efficiency results from the number N of remainders in each of the two domains, often termed the number of DFT or FFT points or the size of the DFT or FFT, being highly composite. One difference between the DFT and FFT then is that while one might speak of a DFT even when N is prime, for example, there is little point in speaking of an FFT in that case, as the FFT degenerates then to an as-written DFT computation.

An *exponentially modulated filter bank* [1, 2] is a bank of filters each of which has a frequency response that is a frequency-shifted version of some prototype response. The frequency shifts are spaced uniformly across one period of the frequency responses using the complex exponentials of a DFT, typically realized with an FFT. The application most familiar to radar engineers is a Doppler filter bank. (The difference between an exponentially modulated filter bank and a cosine-modulated filter bank is the difference respectively between shifting an $f = 0$ passband in the two-sided frequency response to $f = f_0$ and shifting it instead to $f = \pm f_0$.) Because the exponentially modulated filter bank uses frequency responses that are not symmetric about $f = 0$, it is naturally associated with complex signals.

DFT beamsteering uses an exponentially modulated bank of filters in spatial frequency—each spatial frequency of interest corresponds to a plane-wave direction of arrival or DOA—to provide a narrowband digital receive array with many simultaneous beams such as would be needed, for example, certain radar designs [3]. That beamsteering can be viewed as filtering should be no great surprise, given that, as is widely appreciated, an array antenna with identical elements (surrounded by "guard elements" as necessary to ensure identical embedded element patterns) spaced uniformly along a line has an array factor, the contribution to the array pattern of the array's geometry, weighting, and phasing, that is essentially identical in its mathematics to the frequency response of an FIR digital filter. (This parallel is explored further below.)

Tradition favors referring to FFT beamsteering rather than DFT beamsteering, and because the FFT is generally used in implementation this is certainly acceptable. (Occasionally DFT beamsteering is referred to as FFT beam*forming*, but this is simply incorrect.) Here, however, the FFT terminology is reserved for a particular implementation strategy.

DFT beamsteering is not a new idea but is widely appreciated in the signal-processing community, where it is a classic application of the exponentially modulated filter bank. Even there, however, misconceptions are common regarding what an DFT-beamsteering system can and cannot do.

FALSE: DFT beamsteering can only be used on 1D arrays and on 2D arrays with elements laid out on a rectangular grid. CORRECTION: It can easily be used on the triangular grid and in fact on any lattice grid, *i.e.* one comprising integer-weight linear combinations of a pair of linearly independent array-plane basis vectors.

FALSE: The use of DFT beamsteering precludes the use of tapered array factors, and one must live with a high-sidelobe array factor resembling a sinc function. CORRECTION: The choice of the prototype array factor that is shifted by the filter bank to the various beam positions is not restricted. Its design and its steering with the DFT are completely separate matters mathematically. Neither limits the other.

Manuscript approved May 21, 2007.

FALSE: DFT beamsteering spaces beams uniformly in azimuth and/or elevation angle. CORRECTION: It spaces beams uniformly in the direction cosines. Said another way, unit vectors in the beam pointing directions project onto the array plane as a uniform grid. The grid need not be rectangular with rows and columns, but it must be a lattice grid related to the element-location lattice grid in a disciplined way.

FALSE: The number of points in the DFT must be equal to the number of elements. CORRECTION: The array factor as a function of the projection of the DOA unit vector on the array plane (or line)—this is a normalized array-plane spatial frequency—is periodic (in a generalized lattice-related way), and the number of points in the DFT is the number of beams created per period. It need not be related to the number of elements in the array. It can be larger or smaller.

FALSE: The number of FFT points, and hence the number of beams per period of spatial frequency, must be a power of two or a power of two in each direction. CORRECTION: For FFT-style computational efficiency it does need to be highly composite (in a specific matrix sense to be explored later), but it need not be a power of two.

FALSE: The 2D FFT that steers a planar array in both directions is always computed as many 1D row-wise FFT's and column-wise FFT's. CORRECTION: Sometimes it can be. But more choices of beamsteering grids are available if this restriction is abandoned.

The purposes of this paper then are

1. to derive, in Section 2, a general approach to beamsteering using a DFT with a matrix size parameter for an array with identical elements located on an arbitrary planar lattice,

2. to present, in Section 3, system-design examples of the use of this DFT beamsteering approach that represent the variety of useful possibilities in enough detail to enable the determined system designer to address her own design needs, and

3. to present, in Section 4, an generalized FFT algorithm that can be used to realize this beamsteering system by efficiently computing the matrix-size DFT.

This FFT with a matrix size parameter appears to be new to the engineering application community, though mathematically it represents only a specific and relatively simple case from the well-established and much more general area of fast Fourier transforms on groups.

Readers only marginally motivated to face the mathematics are encouraged to look ahead to the concluding remarks of Section 5 before proceeding.

Notational conventions. Matrices and vectors are bold, while scalars are not. Dimensionless unit vectors wear pointy hats, $\widehat{\mathbf{x}}$, because all they do is point, and other vectors may be dimensioned or not. Matrices are uppercase; vectors are lowercase. Notation $|\mathbf{N}|$ with $\mathbf{N}$ a square matrix will mean $|\det(\mathbf{N})|$. Vectors conveying position information are column vectors, while vectors conveying spatial-frequency information are row vectors—different orientations for different Fourier domains. Variables of these spatial and spatial-frequency types are in bold Roman fonts with the letter $\mathbf{k}$ favored for spatial frequencies. Orthogonal projection onto the array plane, or indeed simple *a priori* restriction to the array plane, is denoted using an underscore $\underline{\mathbf{k}}$ that is as flat as that plane, so underlining and a hat together $\widehat{\underline{\mathbf{k}}}$ denote the projection into the array plane of a nonunderlined unit vector $\widehat{\mathbf{k}}$. Position and spatial-frequency vectors, whether or not confined to the array plane, are

three-vectors. A bold italic font is used for two-vectors, and the usual letters are favored when they are integer valued: $\boldsymbol{n}$. As is the custom in mathematics the set of all integers is denoted by $\mathbb{Z}$ for "zahlen," the German verb "to count." Modifying the usual custom here with vertical and horizontal lines, the universal sets of two-vectors of integers are denoted with

$$\begin{aligned}\not{\mathbb{Z}}^2 &\triangleq \left\{ \begin{bmatrix} n_1 \\ n_2 \end{bmatrix} \;: n_1 \in \mathbb{Z}, n_2 \in \mathbb{Z} \right\} \\ \mathbb{Z}^2 &\triangleq \left\{ [n_1, n_2] : n_1 \in \mathbb{Z}, n_2 \in \mathbb{Z} \right\}.\end{aligned}$$

If $\mathcal{V}$ is a set of numbers or vectors then

$$\begin{aligned}\mathbf{L}\mathcal{V} &\triangleq \{\mathbf{L}\boldsymbol{v} : \boldsymbol{v} \in \mathcal{V}\} \\ \mathcal{V}\mathbf{R} &\triangleq \{\boldsymbol{v}\mathbf{R} : \boldsymbol{v} \in \mathcal{V}\} \\ \mathcal{V} + \boldsymbol{a} = \boldsymbol{a} + \mathcal{V} &\triangleq \{\boldsymbol{a} + \boldsymbol{v} : \boldsymbol{v} \in \mathcal{V}\}\end{aligned}$$

so that, for example, $2\mathbb{Z}$ and $1 + 2\mathbb{Z}$ are the sets of even and odd integers respectively.

Summations shown without explicit ranges for their indices are doubly infinite on each integer, so if $\boldsymbol{n} = \left[\begin{smallmatrix} n_1 \\ n_2 \end{smallmatrix}\right]$ for example,

$$\sum_{\boldsymbol{n}} \quad \text{means} \quad \sum_{n_1=-\infty}^{\infty} \sum_{n_2=-\infty}^{\infty} .$$

Integral signs without limits denote integration from $-\infty$ to ∞.

Primes are not transposes or derivatives but simply naming tags to distinguish closely related variables, and the primed variable may be introduced first to reserve the unprimed one for an important later use.

Technical matters referred to "the Allerton paper [4]" are really referred to [4] and to a much-improved journal version together. The latter, currently under preparation, will be the final, authoritative version, and [4] should be considered a draft with occasional errors.

2 Multi-beam steering with a matrix-size DFT

This discussion is divided into three parts. The third leans heavily on the first two, which are independent of each other.

Section 2.1 develops, as preliminary mathematical background, the idea of division with remainder of an integer vector by a nonsingular square integer matrix and the related idea of taking an integer vector modulo such a matrix. This background discussion is of significant length but is unavoidable, as these concepts lie at the heart of the entire paper.

Section 2.2 develops from first principles and the usual array factor and beam sum for an array with identical elements laid out on a 2D lattice. What is new here is not the basic result but the approach, for confidence that all the signs and such are correct, and the specific formulation, for elegance and generality.

Section 2.3 then formulates multi-beam steering using a DFT with a matrix size parameter, which approach is apparently new.

Choosing the grid of points onto which beams are steered in normalized spatial frequency (the array-plane projection of the DOA unit vector) is largely deferred to Section 3, where it will be explored through examples.

2.1 Remaindered division of an integer vector by a square integer matrix

Engineer: "You can do *that*???"

Mathematician: "Of course you can do that... you mean someone cares?"

In anticipation of the development to follow of DFT beamsteering, here integer remainders modulo-N are generalized to integer-vector remainders modulo-$\mathbf{N}$, where $\mathbf{N}$ is a nonsingular 2×2 integer matrix. This author has seen nothing published on the notions presented subsequently on actually computing a vector modulo a matrix, but the material on remaindered division discussed first is known to every graduate student of mathematics using somewhat different terminology, give or take various details, and is a straightforward specialization to lattices of coset decomposition, one of the most elementary parts of group theory at the senior undergraduate level [5].

The simple decomposition in which we are interested here can be written in terms of either row or column vectors. Using $\boldsymbol{n}''$, $\boldsymbol{n}'$, and $\boldsymbol{n}$ from the set $\mathbb{Z}^2$ of integer column two-vectors and $\boldsymbol{k}''$, $\boldsymbol{k}'$, and $\boldsymbol{k}$ from the set $\mathbb{Z}^2$ of integer row two-vectors,

$$\boldsymbol{n}'' = \mathbf{N}\boldsymbol{n}' + \boldsymbol{n} \tag{1}$$

$$\boldsymbol{k}'' = \boldsymbol{k}'\mathbf{N} + \boldsymbol{k}. \tag{2}$$

Each of these expresses a kind of division by matrix $\mathbf{N}$, where $\boldsymbol{n}''$ and $\boldsymbol{k}''$ are the dividends, $\mathbf{N}$ is the divisor, $\boldsymbol{n}'$ and $\boldsymbol{k}'$ are the quotients, and $\boldsymbol{n}$ and $\boldsymbol{k}$ are the remainders.

If the quantities in (1) were all scalars with $\mathbf{N} = 4$, for example, the most common convention would be to require remainders in the set $\{0, 1, 2, 3\}$. Each divisor could then be computed with floating-point division followed by a floor operation, and the rest of the decomposition would follow. But of course unambiguous results would also obtain if instead we required remainders in $\{-2, -1, 0, 1\}$, which decomposition could be computed by obtaining the divisor with floating-point division followed by rounding to the nearest integer. These are just two of many possible remainder sets, any of which would result in unique decompositions of an integer in terms of an integer quotient and an integer remainder. If we relax the "division" interpretation and require only that the decomposition be unique, we can take the remainder set to be $\{1, 2, 3, 4\}$ or $\{-2, 1, 4, 7\}$, for example. What matters is that there are four remainders in the set of allowed possibilities and that no two of them differ by a multiple of four, because it is these conditions that make the decomposition both assuredly possible and unique.

We will ultimately find a way below to arrive at decompositions (1) and (2) with

$$\boldsymbol{n} \in \left[\mathbb{Z}^2/\mathbf{N}\mathbb{Z}^2\right] \subset \mathbb{Z}^2$$
$$\boldsymbol{k} \in \left[\mathbb{Z}^2/\mathbb{Z}^2\mathbf{N}\right] \subset \mathbb{Z}^2,$$

where the notation $\left[\mathbb{Z}^2/\mathbf{N}\mathbb{Z}^2\right]$ is from elementary group theory and in the present context denotes an arbitrary but fixed set of $|\mathbf{N}|$ column two-vectors no two members of which differ by a vector of form $\mathbf{N}\boldsymbol{n}'$, a right integer column-vector multiple of matrix $\mathbf{N}$. Similarly, $\left[\mathbb{Z}^2/\mathbb{Z}^2\mathbf{N}\right]$ denotes an arbitrary but fixed set of $|\mathbf{N}| \triangleq |\det(\mathbf{N})|$ integer row two-vectors no two members of which differ by a vector of form $\boldsymbol{k}'\mathbf{N}$, a left integer row-vector multiple of matrix $\mathbf{N}$. We can think of $\left[\mathbb{Z}^2/\mathbf{N}\mathbb{Z}^2\right]$ and $\left[\mathbb{Z}^2/\mathbb{Z}^2\mathbf{N}\right]$ as our sets respectively of allowable column- and row-vector remainders modulo-$\mathbf{N}$.

We can see how the defining conditions on $\left[\mathbb{Z}^2/\mathbf{N}\mathbb{Z}^2\right]$ guarantee the uniqueness of decomposition (1) by supposing for the moment that uniqueness does not always hold and looking at what is implied when it does not. It would certainly imply the existence, for at least one choice of $\boldsymbol{n}''$, of two distinct decompositions, so that $\boldsymbol{n}'' = \mathbf{N}\boldsymbol{n}_1' + \boldsymbol{n}_1 = \mathbf{N}\boldsymbol{n}_2' + \boldsymbol{n}_2$, from which it would in turn follow that

$$\mathbf{N}(\boldsymbol{n}_1' - \boldsymbol{n}_2') = \boldsymbol{n}_2 - \boldsymbol{n}_1. \tag{3}$$

But if it were so, would $\boldsymbol{n}_2 = \boldsymbol{n}_1$ or not? Matrix $\mathbf{N}$ is nonsingular, so (3) says that $\boldsymbol{n}_2 = \boldsymbol{n}_1$ would be possible only if $\boldsymbol{n}'_1 = \boldsymbol{n}'_2$, in which case the two decompositions would not be distinct after all. But (3) also says that $\boldsymbol{n}_2 \neq \boldsymbol{n}_1$ would only be possible if the two remainders differed by a right integer column-vector multiple of matrix $\mathbf{N}$. As we have disallowed this possibility, nonuniqueness would imply the impossible, that neither $\boldsymbol{n}_2 = \boldsymbol{n}_1$ nor $\boldsymbol{n}_2 \neq \boldsymbol{n}_1$. The decomposition is therefore unique. We have proved it "by contradiction."

But how do we know that decomposition (1) is even possible, that such a combination of an $\boldsymbol{n}'$ and an $\boldsymbol{n}$ exists for every possible $\boldsymbol{n}''$? Vector $\boldsymbol{n}'$ can be any element of $\mathbb{Z}^2$, and the possible values for it therefore occur with unit density in the plane. The set of possible values for $\mathbf{N}\boldsymbol{n}'$ then, which set we denote $\mathbf{N}\mathbb{Z}^2$, has density $\frac{1}{|\mathbf{N}|}$ in the plane. For each of these possible values, the sum $\mathbf{N}\boldsymbol{n}' + \boldsymbol{n}$ can reach $|\mathbf{N}|$ possible totals, so the density of these totals in the plane is $\frac{1}{|\mathbf{N}|} \times |\mathbf{N}| = 1$ provided only that all such totals are distinct, a requirement we know is met by the uniqueness argument above. Therefore, the density of totals in the plane is the same as the density of values $\boldsymbol{n}''$ that we wish to express as totals. There are just enough totals to go around, and every vector $\boldsymbol{n}''$ has a decomposition (1).

Basically then, decomposition (1) makes $\boldsymbol{n}''$ the sum of a multiple of matrix $\mathbf{N}$ and any of $|\mathbf{N}|$ possible remainder vectors. It is similar for decomposition (2) and $\boldsymbol{k}''$.

A vector modulo a matrix. There is a particular choice of remainder set $\left[\mathbb{Z}^2/\mathbf{N}\mathbb{Z}^2\right]$, call it $\left\{\begin{smallmatrix}\text{mod-}\mathbf{N}\\ \text{columns}\end{smallmatrix}\right\}$, that will make decomposition $\boldsymbol{n}'' = \mathbf{N}\boldsymbol{n}' + \boldsymbol{n}$ simple and easy to mechanize. In other words, it will allow us to straightforwardly compute a value

$$\boldsymbol{n} = \boldsymbol{n}'' \bmod \mathbf{N} \in \left\{\begin{smallmatrix}\text{mod-}\mathbf{N}\\ \text{columns}\end{smallmatrix}\right\}$$

that will result in $\boldsymbol{n}'' = \mathbf{N}\boldsymbol{n}' + \boldsymbol{n}$ for some $\boldsymbol{n}' \in \mathbb{Z}^2$. The approach exactly parallels the computation of $n \bmod 4$ as $4\,\mathrm{frac}(n/4)$, where function $\mathrm{frac}(x) \triangleq x - \lfloor x \rfloor$ returns the fractional part of its argument.

Begin by left multiplying relationship $\boldsymbol{n}'' = \mathbf{N}\boldsymbol{n}' + \boldsymbol{n}$ by $\mathbf{N}^{-1}$ and taking the fractional part of each vector componentwise to obtain

$$\mathrm{frac}(\mathbf{N}^{-1}\boldsymbol{n}'') = \mathrm{frac}(\boldsymbol{n}' + \mathbf{N}^{-1}\boldsymbol{n}) = \mathrm{frac}(\mathbf{N}^{-1}\boldsymbol{n}). \tag{4}$$

On the right we can further observe that

$$\mathrm{frac}(\mathbf{N}^{-1}\boldsymbol{n}) = \mathbf{N}^{-1}\boldsymbol{n} \tag{5}$$

whenever each element of vector $\mathbf{N}^{-1}\boldsymbol{n}$ is nonnegative but less than unity, that is, whenever $\mathbf{N}^{-1}\boldsymbol{n} \in \square \triangleq [0,1) \times [0,1)$, the unit square closed on two of its four sides. Let us tentatively take as set $\left\{\begin{smallmatrix}\text{mod-}\mathbf{N}\\ \text{columns}\end{smallmatrix}\right\}$ those values of $\boldsymbol{n}$ for which the latter condition holds or

$$\left\{\begin{smallmatrix}\text{mod-}\mathbf{N}\\ \text{columns}\end{smallmatrix}\right\} \triangleq \{\boldsymbol{n} \in \mathbb{Z}^2 \colon \mathbf{N}^{-1}\boldsymbol{n} \in \square\}. \tag{6}$$

While this definition will be useful formally below, an alternative, equivalent version

$$\left\{\begin{smallmatrix}\text{mod-}\mathbf{N}\\ \text{columns}\end{smallmatrix}\right\} = \{\boldsymbol{n} \in \mathbb{Z}^2 \colon \boldsymbol{n} \in \mathbf{N}\,\square\}. \tag{7}$$

will be more useful for visualization later.

Before using definition (6), we must test it against the two requirements every remainder set $\left[\mathbb{Z}^2/\mathbf{N}\mathbb{Z}^2\right]$ must meet. First, let us determine the number of vectors in $\left\{\begin{smallmatrix}\text{mod-}\mathbf{N}\\ \text{columns}\end{smallmatrix}\right\}$ as defined this way.

Every real number can be written as the sum of its integer and fractional parts, so the real plane, the set of real column two-vectors, can written as the disjoint union

$$\bigcup_{\boldsymbol{n}'\in\mathbb{Z}^2}^{\text{disjoint}} \left(\boldsymbol{n}'+\square\right).$$

and in addition we can certainly intersect the real plane with its subset $\mathbf{N}^{-1}\mathbb{Z}^2$ without changing the latter, so

$$\mathbf{N}^{-1}\mathbb{Z}^2 = \left(\mathbf{N}^{-1}\mathbb{Z}^2\right) \cap \left(\bigcup_{\boldsymbol{n}'\in\mathbb{Z}^2}^{\text{disjoint}} \left(\boldsymbol{n}'+\square\right)\right) = \bigcup_{\boldsymbol{n}'\in\mathbb{Z}^2}^{\text{disjoint}} \left(\left(\mathbf{N}^{-1}\mathbb{Z}^2\right) \cap \left(\boldsymbol{n}'+\square\right)\right). \tag{8}$$

But $\mathbb{Z}^2 = \mathbf{N}\boldsymbol{n}'+\mathbb{Z}^2$, so on the right above

$$\left(\mathbf{N}^{-1}\mathbb{Z}^2\right) \cap \left(\boldsymbol{n}'+\square\right) = \left(\mathbf{N}^{-1}(\mathbf{N}\boldsymbol{n}'+\mathbb{Z}^2)\right) \cap \left(\boldsymbol{n}'+\square\right) = \left(\boldsymbol{n}'+\mathbf{N}^{-1}\mathbb{Z}^2\right) \cap \left(\boldsymbol{n}'+\square\right).$$

The set on the right has the same size as $\left(\mathbf{N}^{-1}\mathbb{Z}^2\right) \cap \square$, because the number of points contained in the intersection is not affected by identical translation of its two components. The sets in the disjoint union in (8) are therefore of identical size, which size must then be just the density per unit area of the left side of (8) or $|\mathbf{N}|$. So this is then the size of $\left\{{\text{mod-}\mathbf{N} \atop \text{columns}}\right\} = \left(\mathbf{N}^{-1}\mathbb{Z}^2\right) \cap \square$ as well, just as it should be.

Second, suppose $\boldsymbol{n}_1, \boldsymbol{n}_2 \in \left\{{\text{mod-}\mathbf{N} \atop \text{columns}}\right\}$ as tentatively defined, and suppose $\boldsymbol{n}_1 - \boldsymbol{n}_2 = \mathbf{N}\boldsymbol{\ell}$ for some nonzero $\boldsymbol{\ell} \in \mathbb{Z}^2$ so that $\mathbf{N}^{-1}\boldsymbol{n}_1 - \mathbf{N}^{-1}\boldsymbol{n}_2 = \boldsymbol{\ell}$. But each of $\mathbf{N}^{-1}\boldsymbol{n}_1$ and $\mathbf{N}^{-1}\boldsymbol{n}_2$ is in $\square$, so the only difference between them that is possible that is integer-valued in both components is zero: $\boldsymbol{n}_1 = \boldsymbol{n}_2$. We have proved by contradiction that no two elements of $\left\{{\text{mod-}\mathbf{N} \atop \text{columns}}\right\}$ differ by a right integer column-vector multiple of matrix $\mathbf{N}$.

We can therefore formally adopt (6) as the definition of $\left[\mathbb{Z}^2/\mathbf{N}\mathbb{Z}^2\right]$, as (6) meets both requirements, and accomplish decomposition $\boldsymbol{n}''= \mathbf{N}\boldsymbol{n}' + \boldsymbol{n}$ with $\boldsymbol{n}' \in \mathbb{Z}^2$ and $\boldsymbol{n} \in \left\{{\text{mod-}\mathbf{N} \atop \text{columns}}\right\}$ by noting that (4) and (5) now give $\mathbf{N}^{-1}\boldsymbol{n} = \operatorname{frac}(\mathbf{N}^{-1}\boldsymbol{n}) = \operatorname{frac}(\mathbf{N}^{-1}\boldsymbol{n}'')$, from which it follows that $\boldsymbol{n} = \boldsymbol{n}'' \bmod \mathbf{N}$, where

$$\boldsymbol{n}'' \bmod \mathbf{N} \triangleq \mathbf{N}\operatorname{frac}(\mathbf{N}^{-1}\boldsymbol{n}''). \tag{9}$$

Geometric interpretation of the mod $\mathbf{N}$ remainder sets. Suppose, for example, that

$$\mathbf{N} = \begin{bmatrix} 0 & -2 \\ -1 & 1 \end{bmatrix}.$$

The determination of $\left\{{\text{mod-}\mathbf{N} \atop \text{columns}}\right\}$ (resp. $\left\{{\text{mod-}\mathbf{N} \atop \text{rows}}\right\}$) according to definition (7) (resp. definition (12) below) are shown on the left (resp. right) in Figure 1. The components of that figure are summarized in the center (resp. right) column of this table.

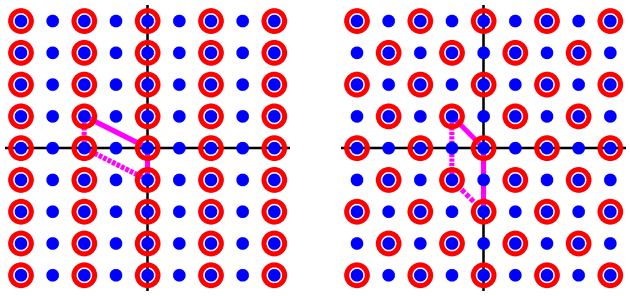


Figure 1: Left and right plot respectively: for $\mathbf{N} = \left[\begin{smallmatrix} 0 & -2 \\ -1 & 1 \end{smallmatrix}\right]$ determine remainder sets

$$\left\{\begin{smallmatrix}\text{mod-}\mathbf{N}\\ \text{columns}\end{smallmatrix}\right\} = \left\{ \left[\begin{smallmatrix} 0 \\ 0 \end{smallmatrix}\right], \left[\begin{smallmatrix} -1 \\ 0 \end{smallmatrix}\right] \right\}$$

$$\left\{\begin{smallmatrix}\text{mod-}\mathbf{N}\\ \text{rows}\end{smallmatrix}\right\} = \left\{ [0 \quad 0], [0 \ -1] \right\}.$$

graphically using (7) and (12).

	left	right
dots	$\mathbb{Z}^2$	$\mathbb{Z}^2$
circled dots	$\mathbf{N}\mathbb{Z}^2$	$\mathbb{Z}^2\mathbf{N}$
transformed unit square	$\mathbf{N}\,\square$	$\square\,\mathbf{N}$
in the semi-open **parallelogram**	$\left\{\begin{smallmatrix}\text{mod-}\mathbf{N}\\ \text{columns}\end{smallmatrix}\right\}$	$\left\{\begin{smallmatrix}\text{mod-}\mathbf{N}\\ \text{rows}\end{smallmatrix}\right\}$

Dots depict $\mathbb{Z}^2$ (resp. $\mathbb{Z}^2$), and **circled dots** depict its subset $\mathbf{N}\mathbb{Z}^2$ (resp. $\mathbb{Z}^2\mathbf{N}$), which comprises the sums of the integer-weighted columns (resp. rows) of $\mathbf{N}$. If $\mathbb{Z}^2$ is replaced in $\mathbf{N}\mathbb{Z}^2$ (resp. $\mathbb{Z}^2$ is replaced in $\mathbb{Z}^2\mathbf{N}$) with the set $\left\{\left[\begin{smallmatrix} 0 \\ 0 \end{smallmatrix}\right], \left[\begin{smallmatrix} 0 \\ 1 \end{smallmatrix}\right], \left[\begin{smallmatrix} 1 \\ 0 \end{smallmatrix}\right], \left[\begin{smallmatrix} 1 \\ 1 \end{smallmatrix}\right]\right\}$ of the four column vectors (resp. set $\{[0\ 0], [1\ 0], [0\ 1], [1\ 1]\}$ of row vectors), the result is the four corners of **transformed unit square** $\mathbf{N}\,\square$ (resp. $\square\,\mathbf{N}$) in definition (7) (resp. definition (12) below). If that **parallelogram** boundary is taken as closed at the origin and on the edges adjacent to the origin and open on other edges and corners, then the **points** of $\mathbb{Z}^2$ (resp. $\mathbb{Z}^2$) that the **parallelogram** contains are precisely the vectors in

$$\left\{\begin{smallmatrix}\text{mod-}\mathbf{N}\\ \text{columns}\end{smallmatrix}\right\} = \left\{ \left[\begin{smallmatrix} 0 \\ 0 \end{smallmatrix}\right], \left[\begin{smallmatrix} -1 \\ 0 \end{smallmatrix}\right] \right\} \qquad \left(\text{resp. } \left\{\begin{smallmatrix}\text{mod-}\mathbf{N}\\ \text{rows}\end{smallmatrix}\right\} = \{[0 \quad 0], [0 \ -1]\}\right). \tag{10}$$

The **parallelogram** tiles the plane when translated by elements of $\mathbf{N}\mathbb{Z}^2$ (resp. $\mathbb{Z}^2\mathbf{N}$) and so can be taken as a period of the latter subset of the plane. When $\mathbf{N}\mathbb{Z}^2$ (resp. $\mathbb{Z}^2\mathbf{N}$) is considered instead as a subset of $\mathbb{Z}^2$ (resp. $\mathbb{Z}^2$), the remainder set $\left\{\begin{smallmatrix}\text{mod-}\mathbf{N}\\ \text{columns}\end{smallmatrix}\right\}$ (resp. $\left\{\begin{smallmatrix}\text{mod-}\mathbf{N}\\ \text{rows}\end{smallmatrix}\right\}$) can be taken as its period. This corresponds exactly to the universality and uniqueness of decomposition (1) (resp. (2)). In Figure 1 the density of $\mathbb{Z}^2$ in the plane is twice that of its subset $\mathbf{N}\mathbb{Z}^2$, so a period of this $\mathbf{N}\mathbb{Z}^2$ must contain exactly two points of $\mathbb{Z}^2$ and the size of the remainder set is determined. In general there are an infinite number of choices for a plane-tiling period, not necessarily involving parallelograms or even a continuous region, and these choices exactly yield the $\left[\mathbb{Z}^2/\mathbf{N}\mathbb{Z}^2\right]$ (resp. $\left[\mathbb{Z}^2/\mathbb{Z}^2\mathbf{N}\right]$) alternatives to $\left\{\begin{smallmatrix}\text{mod-}\mathbf{N}\\ \text{columns}\end{smallmatrix}\right\}$ (resp. $\left\{\begin{smallmatrix}\text{mod-}\mathbf{N}\\ \text{rows}\end{smallmatrix}\right\}$). Here any combination of one (uncircled) **dot** and one **circled dot** would be allowable as a remainder set $\left[\mathbb{Z}^2/\mathbf{N}\mathbb{Z}^2\right]$ (resp. $\left[\mathbb{Z}^2/\mathbb{Z}^2\mathbf{N}\right]$).

Computing mod N safely. The graphical procedure just illustrated is adequate for simple examples, either to determine the entire remainder set or to test a particular vector for membership in that set. (Is it in the **parallelogram**?) But points can, as is the case in Figure 1, lie exactly on the open boundaries of the **parallelogram**. This invites problems with computational noise when these procedures are automated. Can this issue be reliably sidestepped somehow in practice?

Yes, we can in fact easily make the matlab computation of (9) fast and robust and in no danger of suffering edge effects in the frac operation or from numerical noise in the inverse. The frac operation in $\mathrm{frac}(\mathbf{N}^{-1}\boldsymbol{n}'')$ can be viewed as a componentwise modulo-one operation on real numbers,

and we can scale that operation's arguments up by $|\mathbf{N}|$ and its output down by $|\mathbf{N}|$ to compute the frac operation as

$$\mathrm{frac}(\mathbf{N}^{-1}\boldsymbol{n}'') = \frac{1}{|\mathbf{N}|}\Big(|\mathbf{N}|\,\mathbf{N}^{-1}\boldsymbol{n}'' \bmod |\mathbf{N}|\Big)$$

using a componentwise mod operation. Product $|\mathbf{N}|\,\mathbf{N}^{-1}$ contains integers only and can be rounded to eliminate numerical noise and thereby guarantee that the mod operation has a left argument that is numerically an integer vector. Using `npp` for $\boldsymbol{n}''$in matlab then, we can describe the computation $\boldsymbol{n} = \boldsymbol{n}'' \bmod \mathbf{N}$ using (9) with the code

```
function n=columnmod(npp,N)
  adN=abs(det(N));
  n=(N*mod(round(adN*inv(N))*npp,adN))/adN;
```

The final, integer division is remainderless because each component of the numerator is a multiple of the denominator. To see this, note that the mod operation on the last line has the effect of offsetting its left argument by some vector $\mathbf{c}|\mathbf{N}|$ where $\mathbf{c} \in \mathbb{Z}^2$. The numerator is therefore

$$\mathbf{N}\Big(|\mathbf{N}|\mathbf{N}^{-1}\boldsymbol{n}'' + \mathbf{c}|\mathbf{N}|\Big) = (\boldsymbol{n}'' + \mathbf{N}\mathbf{c})|\mathbf{N}|. \tag{11}$$

There are analogous definitions and computations for row vectors to be decomposed according to (2). In this case, and for convenience now taking $\square$ to comprise row vectors,

$$\begin{aligned} \left\{{\text{mod-}\mathbf{N} \atop \text{rows}}\right\} &\triangleq \{\boldsymbol{k} \in \mathbb{Z}^2 \colon \boldsymbol{k} \in \square\,\mathbf{N}\} \\ \boldsymbol{k} &= \boldsymbol{k}'' \bmod \mathbf{N} \\ \boldsymbol{k}'' \bmod \mathbf{N} &\triangleq \mathrm{frac}(\boldsymbol{k}''\mathbf{N}^{-1})\mathbf{N} \end{aligned} \tag{12}$$

with matlab definition

```
function k=rowmod(kpp,N)
  adN=abs(det(N));
  k=(mod(kpp*round(adN*inv(N)),adN)*N)/adN;
```

When $\mathbf{N}$ is just a positive scalar integer N, both `columnmod()` and `rowmod()` degenerate correctly to the usual mod-N operation.

Other remainder-set choices. Let remainder set $\big[\mathbb{Z}^2/\mathbf{N}\mathbb{Z}^2\big]$ once again be arbitrary. For any $\boldsymbol{n}'' \in \mathbb{Z}^2$ there is a unique decomposition, according to (1), of the form

$$\boldsymbol{n}'' = \mathbf{N}\boldsymbol{n}' + \boldsymbol{n}, \tag{13}$$

where $\boldsymbol{n}' \in \mathbb{Z}^2$ and $\boldsymbol{n} \in \big[\mathbb{Z}^2/\mathbf{N}\mathbb{Z}^2\big]$. But how do we effect this decomposition for a given $\boldsymbol{n}''$?

We already know there is also a unique decomposition of the form

$$\boldsymbol{n}'' = \mathbf{N}\boldsymbol{n}'_{\text{mod}} + \boldsymbol{n}_{\text{mod}}, \tag{14}$$

where $\boldsymbol{n}_{\text{mod}} = \boldsymbol{n}'' \bmod \mathbf{N}$ and where $\boldsymbol{n}'_{\text{mod}} \in \mathbb{Z}^2$ can then be solved for. Subtracting (13) from (14) and solving for $\boldsymbol{n}$ yields

$$\boldsymbol{n} = \mathbf{N}\boldsymbol{n}'_{\text{offset}} + \boldsymbol{n}_{\text{mod}}, \tag{15}$$

where $\boldsymbol{n}'_{\text{offset}} = \boldsymbol{n}'_{\text{mod}} - \boldsymbol{n}' \in \mathbb{Z}^2$.

Relationship (15) lets each of $\boldsymbol{n}$ and $\boldsymbol{n}_{\text{mod}}$ be determined unambiguously from the other. The simple case is when $\boldsymbol{n}$ is known first, because the uniqueness of decomposition mod $\mathbf{N}$ then immediately yields the second equality in

$$\boldsymbol{n}_{\text{mod}} = \boldsymbol{n}'' \bmod \mathbf{N} = \boldsymbol{n} \bmod \mathbf{N}. \tag{16}$$

When $\boldsymbol{n}_{\text{mod}}$ is known first in (15) instead, $\boldsymbol{n}$ is known as well, in principle, because if we had two possibilities

$$\begin{aligned} \boldsymbol{n}_1 &= \mathbf{N}\boldsymbol{n}'_{\text{offset1}} + \boldsymbol{n}_{\text{mod}} \\ \boldsymbol{n}_2 &= \mathbf{N}\boldsymbol{n}'_{\text{offset2}} + \boldsymbol{n}_{\text{mod}}, \end{aligned}$$

we'd have $\boldsymbol{n}_1 - \boldsymbol{n}_2 = \mathbf{N}(\boldsymbol{n}'_{\text{offset1}} - \boldsymbol{n}'_{\text{offset2}})$, and $\boldsymbol{n}_1$ and $\boldsymbol{n}_2$, both elements of $\left[\mathbb{Z}^2/\mathbf{N}\mathbb{Z}^2\right]$, would differ by a right integer column-vector multiple of matrix $\mathbf{N}$, which is disallowed.

Since the map $\left[\mathbb{Z}^2/\mathbf{N}\mathbb{Z}^2\right] : \boldsymbol{n}_{\text{mod}} \xrightarrow{\text{mod}\,\mathbf{N}} \left\{{\text{mod-}\mathbf{N} \atop \text{columns}}\right\}$ defined by (16) is therefore one-to-one, and since the range $\left\{{\text{mod-}\mathbf{N} \atop \text{columns}}\right\}$ and the image $\left\{\boldsymbol{n} \bmod \mathbf{N} : \boldsymbol{n} \in \left[\mathbb{Z}^2/\mathbf{N}\mathbb{Z}^2\right]\right\}$ of the domain under that map have the same finite size, the map is onto as well and

$$\left\{{\text{mod-}\mathbf{N} \atop \text{columns}}\right\} = \left\{\boldsymbol{n} \bmod \mathbf{N} : \boldsymbol{n} \in \left[\mathbb{Z}^2/\mathbf{N}\mathbb{Z}^2\right]\right\}. \tag{17}$$

Obtaining $\left\{{\text{mod-}\mathbf{N} \atop \text{columns}}\right\}$ in (17) from $\left[\mathbb{Z}^2/\mathbf{N}\mathbb{Z}^2\right]$ in this way is straightforward, but going in the other direction is possible as well. Because $\boldsymbol{n}$ in (15) is uniquely determined by $\boldsymbol{n}_{\text{mod}}$, we can determine it in practice by a lookup table. Equivalently, the lookup table could provide the difference $\boldsymbol{n} - \boldsymbol{n}_{\text{mod}} = \mathbf{N}\boldsymbol{n}'_{\text{offset}}$ of (15) or just $\boldsymbol{n}'_{\text{offset}}$. Expressing the latter as $\boldsymbol{n}'_{\text{offset}} = \texttt{npoffset}(\boldsymbol{n}_{\text{mod}})$ gives, using (15), the matlab computation of $\boldsymbol{n}$ in (13) from $\boldsymbol{n}''$, or `npp` in matlab, as

```
nmod=columnmod(npp,N);
n = N*npoffset(nmod) + nmod;
```

using the vector `columnmod()` function defined previously and predefined lookup-table array `npoffset()`. Once $\boldsymbol{n}$ is thus determined, $\boldsymbol{n}'$can be solved for to complete decomposition (13).

Having shown implicitly that an arbitrary remainder set $\left[\mathbb{Z}^2/\mathbf{N}\mathbb{Z}^2\right]$ can be expressed in the form

$$\left[\mathbb{Z}^2/\mathbf{N}\mathbb{Z}^2\right] = \left\{\mathbf{N}\,\texttt{npoffset}(\boldsymbol{n}_{\text{mod}}) + \boldsymbol{n}_{\text{mod}} : \boldsymbol{n}_{\text{mod}} \in \left\{{\text{mod-}\mathbf{N} \atop \text{columns}}\right\}\right\}, \tag{18}$$

we can easily verify that the converse is true also, that a set of the form on the right in (18) satisfies the defining requirements for a remainder set $\left[\mathbb{Z}^2/\mathbf{N}\mathbb{Z}^2\right]$, no matter what elements of $\mathbb{Z}^2$ are returned by `npoffset()`. First, the set on the right in (18) has $|\mathbf{N}|$ elements, because $\left\{{\text{mod-}\mathbf{N} \atop \text{columns}}\right\}$ does and because adding the $\mathbf{N}\,\texttt{npoffset}(\boldsymbol{n}_{\text{mod}})$ term to $\boldsymbol{n}_{\text{mod}}$ in (18) cannot land the sum on another element of $\left\{{\text{mod-}\mathbf{N} \atop \text{columns}}\right\}$, as elements of the latter cannot differ by such a term. Second, the difference of two elements of the set on the right in (18) takes the form $(\mathbf{N}\,\texttt{npoffset}(\boldsymbol{n}_{\text{mod1}}) + \boldsymbol{n}_{\text{mod1}}) - (\mathbf{N}\,\texttt{npoffset}(\boldsymbol{n}_{\text{mod2}}) + \boldsymbol{n}_{\text{mod2}}) = \mathbf{N}(\texttt{npoffset}(\boldsymbol{n}_{\text{mod1}}) - \texttt{npoffset}(\boldsymbol{n}_{\text{mod1}})) + \boldsymbol{n}_{\text{mod1}} - \boldsymbol{n}_{\text{mod2}}$ and so can't be a right integer column-vector multiple of matrix $\mathbf{N}$ because $\boldsymbol{n}_{\text{mod1}} - \boldsymbol{n}_{\text{mod2}}$ can't. The possible sets $\left[\mathbb{Z}^2/\mathbf{N}\mathbb{Z}^2\right]$ are therefore exactly those of form (18) for the various possible choices of map $\texttt{npoffset} : \left\{{\text{mod-}\mathbf{N} \atop \text{columns}}\right\} \to \mathbb{Z}^2$.

For row vectors matters are analogous. We have

$$\left[\mathbb{Z}^2/\mathbb{Z}^2\mathbf{N}\right] = \left\{\texttt{klookup}(\boldsymbol{k}_{\text{mod}})\,\mathbf{N} + \boldsymbol{k}_{\text{mod}} : \boldsymbol{k}_{\text{mod}} \in \left\{{\text{mod-}\mathbf{N} \atop \text{rows}}\right\}\right\} \tag{19}$$

with the corresponding matlab for individual values:

```
kmod=rowmod(kpp,N);
k = klookup(kmod)*N + kmod;
```

In the other direction,

$$\left\{{\text{mod-}\mathbf{N} \atop \text{rows}}\right\} = \left\{\boldsymbol{k} \bmod \mathbf{N} : \boldsymbol{k} \in \left[\mathbb{Z}^2/\mathbb{Z}^2\mathbf{N}\right]\right\}.$$

2.2 The 2D lattice array from first principles

This development applies to a narrowband array, in which signals are at (near) Fourier radian frequencies $\pm 2\pi c/\lambda$, with λ the positive scalar wavelength of interest.

A hypothetical one-port array. Let's do a thought experiment. Suppose for the moment that our array has but one output port, at the spatial origin and from which we take our output. Of course physically it has other ports, but we can suppose them terminated to take them out of action. In fact we can suppose our one port to be terminated as well, but we will suppose we are looking at the voltage across the termination resistance and taking that as our signal. In other words, we are looking at the output of a single element, embedded in the array.

Now suppose we translate the entire array in position, without changing its orientation in any way, so as to move the one output port from the origin to position $\mathbf{x}$, a column vector, and let us then call that translated port's output signal $s(\mathbf{x}, t)$. We can write this in terms of its spacetime Fourier transform $S(\boldsymbol{\kappa}, f)$, using row vector $\boldsymbol{\kappa}$ as the spatial transform variable, as

$$s(\mathbf{x}, t) = \int \iiint S(\boldsymbol{\kappa}, f)\, \mathrm{e}^{\mathrm{j}2\pi(ft+\boldsymbol{\kappa}\mathbf{x})}\, d\boldsymbol{\kappa}\, df \tag{20}$$

where $d\boldsymbol{\kappa}$ is differential spatial-frequency volume.

How do we interpret Fourier transform $S(\boldsymbol{\kappa}, f)$? Output signal $s(\mathbf{x}, t)$ is necessarily real, and it follows that $S(\boldsymbol{\kappa}, f) = S^*(-\boldsymbol{\kappa}, -f)$. Assuming then that $S(\boldsymbol{\kappa}, 0) = 0$ (no DC output to confuse region-of-integration endpoint matters below), we can show, through an annoying number of small steps beginning with splitting the integral in f, that

$$\begin{aligned}
s(\mathbf{x}, t) &= \int_{-\infty}^{0} \iiint S(\boldsymbol{\kappa}, f)\, \mathrm{e}^{\mathrm{j}2\pi(ft+\boldsymbol{\kappa}\mathbf{x})}\, d\boldsymbol{\kappa}\, df \qquad + \int_{0}^{\infty} \iiint S(\boldsymbol{\kappa}, f)\, \mathrm{e}^{\mathrm{j}2\pi(ft+\boldsymbol{\kappa}\mathbf{x})}\, d\boldsymbol{\kappa}\, df \\
&= \int_{0}^{\infty} \iiint S(-\boldsymbol{\kappa}, -f)\, \mathrm{e}^{-\mathrm{j}2\pi(ft+\boldsymbol{\kappa}\mathbf{x})}\, d\boldsymbol{\kappa}\, df + \int_{0}^{\infty} \iiint S(\boldsymbol{\kappa}, f)\, \mathrm{e}^{\mathrm{j}2\pi(ft+\boldsymbol{\kappa}\mathbf{x})}\, d\boldsymbol{\kappa}\, df \\
&= \int_{0}^{\infty} \iiint S^*(\boldsymbol{\kappa}, f)\, \mathrm{e}^{-\mathrm{j}2\pi(ft+\boldsymbol{\kappa}\mathbf{x})}\, d\boldsymbol{\kappa}\, df \qquad + \int_{0}^{\infty} \iiint S(\boldsymbol{\kappa}, f)\, \mathrm{e}^{\mathrm{j}2\pi(ft+\boldsymbol{\kappa}\mathbf{x})}\, d\boldsymbol{\kappa}\, df \\
&= \int_{0}^{\infty} \iiint \mathrm{Re}\big\{2S(\boldsymbol{\kappa}, f)\, \mathrm{e}^{\mathrm{j}2\pi(ft+\boldsymbol{\kappa}\mathbf{x})}\big\}\, d\boldsymbol{\kappa}\, df \\
&= \int_{0}^{\infty} \iiint \mathrm{Re}\big\{2S(-\boldsymbol{\kappa}_{\text{wavenumber}}, f)\, \mathrm{e}^{\mathrm{j}2\pi(ft-\boldsymbol{\kappa}_{\text{wavenumber}}\mathbf{x})}\big\}\, d\boldsymbol{\kappa}_{\text{wavenumber}}\, df
\end{aligned}$$

This final form looks exactly like the integral traditionally used to combine complex field components, each a propagating plane wave, to form the real field. If our one-port array were a perfect polarized field-measuring device, measuring perhaps the field along some one vector direction, then that interpretation would apply directly. Even though it is not really such a device, it does seem that we are integrating over frequency and wavenumber to combine some sort of propagating complex waves, with the wave at wavenumber vector $-\boldsymbol{\kappa}$ and frequency $f > 0$ having complex amplitude $2S(\boldsymbol{\kappa}, f)$.

It is not necessary to go into the mathematics here, but in fact the difference between the integral in our one-port thought experiment and the traditional field integral is that in our integral the complex amplitude function $2S(\boldsymbol{\kappa}, f)$ governing the amplitudes and phases of the waves has been preprocessed behind the scenes with the embedded element pattern. Because this element preprocessing is linear and both space and time invariant, the Helmholtz relation governing free-space

electromagnetic propagation will apply after element processing just as it does before, and transform variable $\boldsymbol{\kappa}$ will have length $\|\boldsymbol{\kappa}\| = 1/\lambda$ just as a wavenumber vector would. The difference is that here $\boldsymbol{\kappa}$ points, for components having $f > 0$, not in the direction of propagation but in the DOA. We will call $\boldsymbol{\kappa}$ in (20) the *DOA wave vector*.

By the nature of its derivation above transform $S(\boldsymbol{\kappa}, f)$ for $f > 0$ represents the element-preprocessed incoming plane-wave signal at frequency f and wavenumber $-\boldsymbol{\kappa}$. We next derive the actual array output. Comparing the latter with this one-port array output will give us the array factor we seek.

The array factor and beam sum. If we fix $\mathbf{x} = 0$ in (20) the integral gives us the untranslated embedded element output. Let us take that as a reference signal to be improved upon through the creation somehow of a signal

$$X(\lambda\boldsymbol{\kappa}_s, t) \triangleq \int \iiint G_{\text{ideal}}(\boldsymbol{\kappa} - \boldsymbol{\kappa}_s) S(\boldsymbol{\kappa}, f)\, \mathrm{e}^{\mathrm{j}2\pi f t}\, d\boldsymbol{\kappa}\, df. \tag{21}$$

This is just integral (20) with $\mathbf{x} = 0$ and with $S(\boldsymbol{\kappa}, f)$ replaced with $G_{\text{ideal}}(\boldsymbol{\kappa} - \boldsymbol{\kappa}_s)S(\boldsymbol{\kappa}, f)$. A convenient normalization has been adopted for the function's first argument, which is really just a parameter.

A quantity serving the role that $G_{\text{ideal}}(\boldsymbol{\kappa} - \boldsymbol{\kappa}_s)$ does in (21) is traditionally termed an *array factor*. We can suppose that the $\boldsymbol{\kappa}_s = 0$ or *unsteered* array factor $G_{\text{ideal}}(\boldsymbol{\kappa})$ is approximately unity when DOA wave vector $\boldsymbol{\kappa}$ points in some boresight direction and is near zero when it points well away from that direction so that—this is how we express it—array factor $G_{\text{ideal}}(\boldsymbol{\kappa})$ has a *boresight beam*. In our modified combining integral $G_{\text{ideal}}(\boldsymbol{\kappa})$ actually appears translated by $\boldsymbol{\kappa}_s$, so there the beam occurs for $\boldsymbol{\kappa} \approx \boldsymbol{\kappa}_s$ instead of for $\boldsymbol{\kappa} \approx 0$.

How do we create a signal that will be described by such a modified combining integral? What we want is 4D Fourier-pair relationship

$$X(\lambda\boldsymbol{\kappa}_s, t) = \langle \text{mystery function of } \mathbf{x} \text{ and } t\rangle\Big|_{\mathbf{x}=0},$$

$$\langle \text{mystery function of } \mathbf{x} \text{ and } t\rangle \leftrightarrow\ G_{\text{ideal}}(\boldsymbol{\kappa} - \boldsymbol{\kappa}_s)S(\boldsymbol{\kappa}, f),$$

because sampling at the origin in one domain is equivalent to integration in the other. The convolution property of the transform and 1D Fourier pair $\delta(t) \leftrightarrow 1$ then yield

$$\begin{aligned}\langle \text{mystery function of } \mathbf{x} \text{ and } t\rangle &= \big(\delta(t)\langle \text{mystery function of } \mathbf{x}\rangle\big) \star s(\mathbf{x}, t)\\ &\leftrightarrow\ G_{\text{ideal}}(\boldsymbol{\kappa} - \boldsymbol{\kappa}_s)S(\boldsymbol{\kappa}, f),\end{aligned}$$

where $\langle \text{mystery function of } \mathbf{x}\rangle \leftrightarrow G_{\text{ideal}}(\boldsymbol{\kappa} - \boldsymbol{\kappa}_s)$. We can proceed further along these lines by using the modulation property of the transform to write

$$\langle \text{mystery function of } \mathbf{x}\rangle = \mathrm{e}^{\mathrm{j}2\pi\boldsymbol{\kappa}_s\mathbf{x}}\, g_{\text{ideal}}(\boldsymbol{\kappa})\ \leftrightarrow\ G_{\text{ideal}}(\boldsymbol{\kappa} - \boldsymbol{\kappa}_s)$$

with $g_{\text{ideal}}(\boldsymbol{\kappa}) \leftrightarrow G_{\text{ideal}}(\boldsymbol{\kappa})$. Next, since $g_{\text{ideal}}(\mathbf{x})$ is a spatial impulse response we are convolving with the input signal in a sort of spatial filtering operation, let's specialize to the simplest kind of filter to design, a finite-impulse-response or FIR filter: we require that $g_{\text{ideal}}(\mathbf{x})$ comprise a finite number of impulses. For convenience we will make the number formally infinite but with all but a finite number of their areas $\{g_{\boldsymbol{n}''}\}$ nonzero. Then we can write

$$g_{\text{ideal}}(\mathbf{x}) = \sum_{\boldsymbol{n}''} g_{\boldsymbol{n}''}\, \delta(\mathbf{x} - \mathbf{x}_{\boldsymbol{n}''})\ \leftrightarrow\ G_{\text{ideal}}(\boldsymbol{\kappa}). \tag{22}$$

As far as this development is concerned, those weights $\{g_{\boldsymbol{n}''}\}$ that are nonzero might all be unity, as for an untapered beam, or they might be strictly positive, or not, or they might even be complex. It is a system-design issue unrelated to the steering matters at hand.

So far then we have

$$\begin{aligned}
\langle \text{mystery function of } \mathbf{x} \text{ and } t\rangle &= \big(\delta(t)\langle \text{mystery function of } \mathbf{x}\rangle\big) \star s(\mathbf{x},t) \\
&= \big(\delta(t)\,\mathrm{e}^{\mathrm{j}2\pi\boldsymbol{\kappa}_s\mathbf{x}}\, g_{\text{ideal}}(\boldsymbol{\kappa})\big) \star s(\mathbf{x},t) \\
&= \big(\delta(t)\,\mathrm{e}^{\mathrm{j}2\pi\boldsymbol{\kappa}_s\mathbf{x}} \sum_{\boldsymbol{n}''} g_{\boldsymbol{n}''}\,\delta(\mathbf{x}-\mathbf{x}_{\boldsymbol{n}''})\big) \star s(\mathbf{x},t) \\
&= \big(\delta(t) \sum_{\boldsymbol{n}''} \mathrm{e}^{\mathrm{j}2\pi\boldsymbol{\kappa}_s\mathbf{x}_{n''}}\, g_{\boldsymbol{n}''}\,\delta(\mathbf{x}-\mathbf{x}_{\boldsymbol{n}''})\big) \star s(\mathbf{x},t) \\
&= \sum_{\boldsymbol{n}''} \mathrm{e}^{\mathrm{j}2\pi\boldsymbol{\kappa}_s\mathbf{x}_{n''}}\, g_{\boldsymbol{n}''}\, s(\mathbf{x}-\mathbf{x}_{\boldsymbol{n}''},t),
\end{aligned}$$

which when evaluated at $\mathbf{x}=0$ becomes our desired array output

$$X(\lambda\boldsymbol{\kappa}_s,t) = \sum_{\boldsymbol{n}''} \mathrm{e}^{\mathrm{j}2\pi\boldsymbol{\kappa}_s\mathbf{x}_{n''}}\, g_{\boldsymbol{n}''}\, s(-\mathbf{x}_{\boldsymbol{n}''},t). \tag{23}$$

An array output is termed a *beam* or *beam sum* in hopes of a particular functionality.

The array structure must be periodic in a lattice sense. The sum (23) depends on the output of our one-port array when the array is translated by each $-\mathbf{x}_{\boldsymbol{n}''}$, in effect requiring us to have access to many different translated arrays simultaneously. This actually requires only that the original array and the array translated by an arbitrary one of the $-\mathbf{x}_{\boldsymbol{n}''}$ are identical. Then since the array has an output port at the origin, it must have one at $-\mathbf{x}_{\boldsymbol{n}''}$ as well. Both arrays, untranslated and translated, exist simultaneously for our purposes if we observe the signals across both terminations. Taking it further, the array must be physically unchanged by translations from set $\mathcal{X} \triangleq \{-\mathbf{x}_{\boldsymbol{n}''} : \boldsymbol{n}'' \in \mathbb{Z}^2\}$ and must have a port at each element of $\mathcal{X}$.

Now suppose we translate the array twice in succession, first by one vector from $\mathcal{X}$ and then by another. Since neither translation changed the array, and since the net translation is the sum of the two translations, we see that if the array is unchanged by translations from $\mathcal{X}$, then it is in fact unchanged by translations from the algebraic closure of $\mathcal{X}$ under vector addition. We may as well start off by requiring $\mathcal{X}$ to be closed under vector addition, and we can just set $g_{\boldsymbol{n}''}=0$ for any member of $\mathcal{X}$ not actually needed. Since there is an output port at each point in $\mathcal{X}$, and since a port occupies some physical space, the set $\mathcal{X}$ must be discrete, that is, each point in it must be at least some positive minimum distance from all other points. We can assume that set $\mathcal{X}$ contains the origin, because we know there is a port there. If translating by an element of $\mathcal{X}$ did not change the array, then neither does translating by the negative of that element to get back to where we started, so $\mathcal{X}$ is closed under negation. A discrete subset of real Euclidean space that contains the origin and is closed under negation and summation of its vectors is a *lattice* [6].

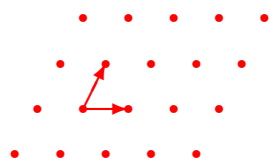
Aside: In this paper a *grid* is just a less formal name for a lattice.

That this array is unchanged by translation by elements of a lattice means it is periodic in structure and has infinite extent, but is it infinite in one, two, or three dimensions? We will, of course, extend the physical structure only some maximum distance out from the origin and just use "guard elements" around the boundary to fool the elements whose outputs we are using into behaving electrically as if they are in an infinite array. But even with this bounding of the array extent, the 3D version appears impractical, because given a plane wave propagating through the

structure, the power out of every port would approach zero because of shadowing "upstream" by an infinite number of other ports, each of which would have a termination consuming power from the wave. The most general structure available to us then is a 2D lattice of elements. The discussion below will assume such a structure and specialize the notation to it, though redoing it all for the uniformly spaced line array, which uses a 1D lattice, would not be difficult.

The element-location lattice. Let $\mathbf{b}_1$ and $\mathbf{b}_2$, each a three-element column vector, be linearly independent, dimensionless *basis vectors*, and then form from them *element-location basis* matrix $\mathbf{B} \triangleq \big[\mathbf{b}_1, \mathbf{b}_2\big]$. The basis terminology is justified because the smallest lattice including these two vectors, in other words the smallest set including these two vectors that is algebraically closed under negation and vector addition, is the set

$$\mathbf{B}\mathbb{Z}^2,$$



the integer-weighted sums of the basis vectors. We can scale this lattice by λ to give its elements units of length[1] and then set location $\mathbf{x}_{\boldsymbol{n}''} = \lambda\mathbf{B}\boldsymbol{n}''$. For notational economy let us further define element output $s_{\boldsymbol{n}''}$, indexed by the two integers of *element index* $\boldsymbol{n}'' \in \mathbb{Z}^2$, to be the signal taken from the output port at nominal *element location* $-\mathbf{x}_{\boldsymbol{n}''} = -\lambda\mathbf{B}\boldsymbol{n}''$so that $s_{\boldsymbol{n}''} = s(-\mathbf{x}_{\boldsymbol{n}''}, t)$. Our signals are functions of time but for economy of notation we will leave the time dependence of $s_{\boldsymbol{n}''}$ implicit as time does not enter into the beamsteering discussion below.

The element locations are all contained in the linear span of $\{\mathbf{b}_1, \mathbf{b}_2\}$, which span is therefore termed the *array plane*. We term $\mathbf{B}\mathbb{Z}^2$ and $\lambda\mathbf{B}\mathbb{Z}^2$ the *normalized* and *unnormalized element-location lattices* respectively.

Normalized variables for the lattice beam sum and array factor. We can use (22) to obtain the unsteered array factor $G_{\text{ideal}}(\boldsymbol{\kappa})$ governing the integral combination in (21). Using $d\mathbf{x}$ for differential spatial volume,

$$\begin{aligned} G_{\text{ideal}}(\boldsymbol{\kappa}) &= \iiint \Big(\sum_{\boldsymbol{n}''} g_{\boldsymbol{n}''}\, \delta(\mathbf{x} - \lambda\mathbf{B}\boldsymbol{n}'')\Big) \mathrm{e}^{-\mathrm{j}2\pi\boldsymbol{\kappa}\mathbf{x}}\, d\mathbf{x} \\ &= \sum_{\boldsymbol{n}''} g_{\boldsymbol{n}''}\, \mathrm{e}^{-\mathrm{j}2\pi\boldsymbol{\kappa}\lambda\mathbf{B}\boldsymbol{n}''} \\ &= G(\boldsymbol{\kappa}\lambda\mathbf{B}) \end{aligned}$$

in terms of a function

$$G(\boldsymbol{f}) \triangleq \sum_{\boldsymbol{n}''} g_{\boldsymbol{n}''}\, \mathrm{e}^{-\mathrm{j}2\pi\boldsymbol{f}\boldsymbol{n}''}, \tag{24}$$

where $\boldsymbol{f}$ is a 2D row-vector normalized frequency. We recognize (24) as the discrete-time Fourier transform in two dimensions of weights $g_{\boldsymbol{n}''}$. Since the number of nonzero weights is finite, it is also the frequency response of an FIR filter with weights $g_{\boldsymbol{n}''}$. Steered array factor $G_{\text{ideal}}(\boldsymbol{\kappa} - \boldsymbol{\kappa}_s)$ now specializes to $G\big((\boldsymbol{\kappa} - \boldsymbol{\kappa}_s)\lambda\mathbf{B}\big)$.

Helmholtz relation $\|\boldsymbol{\kappa}\| = 1/\lambda$ makes normalized DOA wave vector $\boldsymbol{\kappa}\lambda$ a unit vector, and it is natural to make normalized DOA steering vector $\boldsymbol{\kappa}_s\lambda$ a unit vector as well so that the beam of $G((\boldsymbol{\kappa}-\boldsymbol{\kappa}_s)\lambda\mathbf{B})$ points in some physical direction $\boldsymbol{\kappa}_s\lambda$. Let us create notation for these unit vectors[2]

[1] In the Allerton paper [4] the basis itself has length units: the quantity $\lambda\mathbf{B}$ of this paper is the Allerton paper's $\mathbf{B}_c$, the computational representation of the element-location basis $\mathbf{B}$ that plays such a central rôle in the development there.

[2] Expression $\frac{1}{\lambda}\underline{\widehat{\boldsymbol{\kappa}}}$ here is the $\underline{\mathbf{k}}$ of the Allerton paper [4].

and further decompose them into components in and normal to the array plane:

$$\lambda\boldsymbol{\kappa} = \widehat{\boldsymbol{\kappa}} = \underline{\widehat{\boldsymbol{\kappa}}} + \widehat{\boldsymbol{\kappa}}_{\text{normal}}$$
$$\lambda\boldsymbol{\kappa}_s = \widehat{\mathbf{k}}'' = \underline{\widehat{\mathbf{k}}}'' + \widehat{\mathbf{k}}''_{\text{normal}}.$$

Our steered array factor $G\big((\boldsymbol{\kappa}-\boldsymbol{\kappa}_s)\lambda\mathbf{B}\big)$ now becomes $G\big((\underline{\widehat{\boldsymbol{\kappa}}} - \underline{\widehat{\mathbf{k}}}'')\mathbf{B}\big)$, in which the normal components were lost in the product with $\mathbf{B}$, the columns of which are in the array plane. This array factor is often plotted against the *direction cosines*, the components of *normalized array-plane DOA wave vector* $\underline{\widehat{\boldsymbol{\kappa}}}$ in orthogonal "azimuth" and "elevation" directions in the array plane. (Precisely for convenience in plotting, notation for those components will be developed explicitly in Section 3.)

Beam sum (23) can be written now in terms of element-location basis matrix $\mathbf{B}$ and normalized array-plane DOA steering vector $\underline{\widehat{\mathbf{k}}}''$ as

$$X(\lambda\boldsymbol{\kappa}_s) = \sum_{\boldsymbol{n}''} \mathrm{e}^{\mathrm{j}2\pi\underline{\widehat{\mathbf{k}}}''\mathbf{B}\boldsymbol{n}''}\, g_{\boldsymbol{n}''}\, s_{\boldsymbol{n}''} = X(\underline{\widehat{\mathbf{k}}}''). \tag{25}$$

The normal component of the steering vector has been dropped on the right as irrelevant.

A delay interpretation. We can get a check on the formulation of the beam sum by rewriting steered weight $\mathrm{e}^{\mathrm{j}2\pi\underline{\widehat{\mathbf{k}}}''\mathbf{B}\boldsymbol{n}''}g_{\boldsymbol{n}''}$ in (25) as $\mathrm{e}^{-\mathrm{j}2\pi f\tau(\widehat{\mathbf{k}}'',\boldsymbol{n}'')}g_{\boldsymbol{n}''}$, where frequency $f = c/\lambda$ and where delay

$$\tau(\widehat{\mathbf{k}}'',\boldsymbol{n}'') \triangleq -\lambda\widehat{\mathbf{k}}''\mathbf{B}\boldsymbol{n}''/c = -(-\widehat{\mathbf{k}}'')(-\lambda\mathbf{B}\boldsymbol{n}'')/c,$$

a dot product with a unit vector that reveals the delay to be an advance (negative delay, using the first minus sign on the right) of the signal from the element at position $-\lambda\mathbf{B}\boldsymbol{n}''$ by the free-space propagation time across the component of that position along wave-propagation direction $-\widehat{\mathbf{k}}$. This advancing in time counters the time-of-flight delay for the wave to reach element $\boldsymbol{n}''$ relative to when it reaches the spatial origin.

2.3 Many Beams from the DFT

The steering lattice. What makes steering with a DFT possible is also the main limitation of the approach: the family of allowable normalized array-plane DOA steering vectors $\underline{\widehat{\mathbf{k}}}''$ must be chosen in a particular way. Let us begin by attempting to choose it so as to undo or cancel the effect that element-location basis matrix $\mathbf{B}$ has on steering exponential $\mathrm{e}^{\mathrm{j}2\pi\underline{\widehat{\mathbf{k}}}''\mathbf{B}\boldsymbol{n}''}$. One ordinarily cancels a matrix with its inverse, but of course we cannot invert 3×2 matrix $\mathbf{B}$. However, since the basis vectors in $\mathbf{B}$ are linearly independent, Gram matrix $\mathbf{B}^T\mathbf{B}$ is assuredly invertible, and we do therefore have use of the Moore-Penrose pseudoinverse $\mathbf{B}$ that is computed by matlab's `pinv()` function and defined by

$$\mathbf{B}^+ \triangleq (\mathbf{B}^T\mathbf{B})^{-1}\mathbf{B}^T, \tag{26}$$

The usefulness of the pseudoinverse (here) is in the property $\mathbf{B}^+\mathbf{B} = (\mathbf{B}^T\mathbf{B})^{-1}\mathbf{B}^T\mathbf{B} = \mathbf{I}$. So let us set steering vector

$$\underline{\widehat{\mathbf{k}}}'' = -\boldsymbol{k}''\mathbf{N}^{-1}\mathbf{B}^+, \tag{27}$$

where $\mathbf{N}$ is a nonsingular 2×2 integer matrix, the design of which will be explored in Section 3. The quadratic form in steering exponential $\mathrm{e}^{\mathrm{j}2\pi\underline{\widehat{\mathbf{k}}}''\mathbf{B}\boldsymbol{n}''}$ now becomes

$$\begin{aligned}\underline{\widehat{\mathbf{k}}}''\mathbf{B}\boldsymbol{n}'' &= -\boldsymbol{k}''\mathbf{N}^{-1}(\mathbf{B}^T\mathbf{B})^{-1}\mathbf{B}^T\mathbf{B}\boldsymbol{n}'' \\ &= -\boldsymbol{k}''\mathbf{N}^{-1}\boldsymbol{n}'',\end{aligned}$$

and we have not only eliminated $\mathbf{B}$, we have replaced it with $-\mathbf{N}^{-1}$. (The minus sign will prove to be a minor convenience.)

What makes this interesting is the Section 2.1 material on remaindered division by an integer matrix, because substituting decomposition (2) into (27) results in

$$\begin{aligned}\underline{\widehat{\mathbf{k}}}''\mathbf{B}\boldsymbol{n}'' &= -(\boldsymbol{k}'\mathbf{N}+\boldsymbol{k})\mathbf{N}^{-1}\boldsymbol{n}'' \\ &= -\boldsymbol{k}'\boldsymbol{n}''-\boldsymbol{k}\mathbf{N}^{-1}\boldsymbol{n}'',\end{aligned}$$

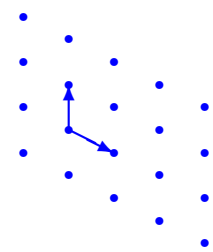
which makes the steering exponential itself just

$$\mathrm{e}^{\mathrm{j}2\pi\underline{\widehat{\mathbf{k}}}''\mathbf{B}\boldsymbol{n}''} = \mathrm{e}^{-\mathrm{j}2\pi\boldsymbol{k}'\boldsymbol{n}''}\mathrm{e}^{-\mathrm{j}2\pi\boldsymbol{k}\mathbf{N}^{-1}\boldsymbol{n}''} = \mathrm{e}^{-\mathrm{j}2\pi\boldsymbol{k}\mathbf{N}^{-1}\boldsymbol{n}''}. \tag{28}$$

The magic here is that all periodicity has disappeared with the factor $\mathrm{e}^{\mathrm{j}2\pi\boldsymbol{k}'\boldsymbol{n}''}$, because any two choices for $\boldsymbol{k}\in\left[\mathbb{Z}^2/\mathbb{Z}^2\mathbf{N}\right]$, say $\boldsymbol{k}_1$ and $\boldsymbol{k}_2$, result in $\mathrm{e}^{-\mathrm{j}2\pi\boldsymbol{k}_1\mathbf{N}^{-1}\boldsymbol{n}''}=\mathrm{e}^{-\mathrm{j}2\pi\boldsymbol{k}_2\mathbf{N}^{-1}\boldsymbol{n}''}$ if and only if $\boldsymbol{k}_1\mathbf{N}^{-1}\boldsymbol{n}''-\boldsymbol{k}_2\mathbf{N}^{-1}\boldsymbol{n}''$ is an integer, and it is an integer for all $\boldsymbol{n}''\in\mathbb{Z}^2$ if and only if $\boldsymbol{k}_1\mathbf{N}^{-1}-\boldsymbol{k}_2\mathbf{N}^{-1}\in\mathbb{Z}^2$. But if $\boldsymbol{k}_1\mathbf{N}^{-1}-\boldsymbol{k}_2\mathbf{N}^{-1}=\boldsymbol{k}_{\text{offset}}\in\mathbb{Z}^2$, then $\boldsymbol{k}_1-\boldsymbol{k}_2=\boldsymbol{k}_{\text{offset}}\mathbf{N}$, which form is disallowed unless $\boldsymbol{k}_1=\boldsymbol{k}_2$. So when decomposition (2) is used for $\boldsymbol{k}''$ in steering-vector choice (27), we can think of $\boldsymbol{k}'$as selecting which steering period we are in and $\boldsymbol{k}$ as selecting a point within that period. We have $|\mathbf{N}|$ steering-vector choices within the steering period through our choice of $\boldsymbol{k}\in\left[\mathbb{Z}^2/\mathbb{Z}^2\mathbf{N}\right]$, so in this context let us term $\mathbf{N}$ the *steering density matrix*.

A very special case is when steering-vector choice (27) places $\boldsymbol{k}''$in $\mathbb{Z}^2\mathbf{N}$, a subset (sublattice in fact) of $\mathbb{Z}^2$. In that case (27) places normalized array-plane DOA steering vectors $\underline{\widehat{\mathbf{k}}}''$ in the set

$$\mathbb{Z}^2\mathbf{B}^+,$$



which comprises integer combinations of the two vectors given by the three-vector rows of $\mathbf{B}^+$. Those rows necessarily lie in the array plane, because definition (26) sets each of those rows to a linear combination of the basis vectors in $\mathbf{B}$. Further, the rows of $\mathbf{B}^+$ are linearly independent, because given any row vector $\boldsymbol{c}$ of coefficients to combine them, $0=\boldsymbol{c}\mathbf{B}^+$ implies $0=\boldsymbol{c}\mathbf{B}^+\mathbf{B}=\boldsymbol{c}$. The rows of 2×3 matrix $\mathbf{B}^+$ therefore form an alternative basis for the array plane, and $\mathbb{Z}^2\mathbf{B}^+$is a lattice, here termed the *boresight replication lattice*. (The name will gain meaning in Section 3.)

Mathematically, $\mathbb{Z}^2\mathbf{B}^+$is the *dual* of lattice $\mathbf{B}\mathbb{Z}^2$(and $\mathbf{B}\mathbb{Z}^2$is the dual of $\mathbb{Z}^2\mathbf{B}^+$), so here let us term $\mathbf{B}^+$the *dual basis* matrix. Further, in (27) the two integers of row *beam index* vector $\boldsymbol{k}''\in\mathbb{Z}^2$ govern the choice of normalized array-plane DOA steering vector $\underline{\widehat{\mathbf{k}}}''$as an integer combination of the two basis vectors given by the three-vector rows of $-\mathbf{N}^{-1}\mathbf{B}^+$. Matrix $\mathbf{N}^{-1}$ is of course invertible, so the rows of $-\mathbf{N}^{-1}\mathbf{B}^+$ are linearly independent and also form a basis. The lattice $\mathbb{Z}^2\mathbf{N}^{-1}\mathbf{B}^+$ of row vectors in the array plane that they specify is the *steering lattice*. The particular form we have given it here will lead to a DFT next and then to an FFT in Section 4.

DFT Beamsteering. Specializing beam sum (25) to use the particular normalized array-plane DOA steering vector $\underline{\widehat{\mathbf{k}}}''$of steering specification (27) yields

$$X_{\boldsymbol{k}''} \triangleq X(-\boldsymbol{k}''\mathbf{N}^{-1}\mathbf{B}^+)$$

$$= \sum_{\boldsymbol{n}''} \mathrm{e}^{\mathrm{j}2\pi \underline{\widehat{\mathbf{k}}}''\mathbf{B}\boldsymbol{n}''} g_{\boldsymbol{n}''}\, s_{\boldsymbol{n}''}$$

$$= \sum_{\boldsymbol{n}''} \mathrm{e}^{-\mathrm{j}2\pi \boldsymbol{k}''\mathbf{N}^{-1}\boldsymbol{n}''} g_{\boldsymbol{n}''}\, s_{\boldsymbol{n}''}. \tag{29}$$

Let us use decomposition (2) for $\boldsymbol{k}''$as discussed above, with remainder set $\left[\mathbb{Z}^2/\mathbb{Z}^2\mathbf{N}\right]$ arbitrary (until the design examples of Section 3), and let us also use decomposition (1) for $\boldsymbol{n}''$ with $\left[\mathbb{Z}^2/\mathbf{N}\mathbb{Z}^2\right] = \left\{\begin{smallmatrix}\text{mod-}\mathbf{N}\\ \text{columns}\end{smallmatrix}\right\}$, a choice that is not necessary but that is convenient. Using (28) and repeating the idea for $\boldsymbol{n}''$,

$$\mathrm{e}^{\mathrm{j}2\pi \underline{\widehat{\mathbf{k}}}''\mathbf{B}\boldsymbol{n}''} = \mathrm{e}^{-\mathrm{j}2\pi \boldsymbol{k}\mathbf{N}^{-1}(\mathbf{N}\boldsymbol{n}'+\boldsymbol{n})} = \mathrm{e}^{-\mathrm{j}2\pi \boldsymbol{k}\boldsymbol{n}'}\mathrm{e}^{-\mathrm{j}2\pi \boldsymbol{k}\mathbf{N}^{-1}\boldsymbol{n}} = \mathrm{e}^{-\mathrm{j}2\pi \boldsymbol{k}\mathbf{N}^{-1}\boldsymbol{n}},$$

using which beam sum (29) becomes

$$X_{\boldsymbol{k}''} = \sum_{\boldsymbol{n}'} \sum_{\boldsymbol{n}\in\left\{\begin{smallmatrix}\text{mod-}\mathbf{N}\\ \text{columns}\end{smallmatrix}\right\}} \mathrm{e}^{-\mathrm{j}2\pi \boldsymbol{k}\mathbf{N}^{-1}\boldsymbol{n}} g_{\mathbf{N}\boldsymbol{n}'+\boldsymbol{n}}\, s_{\mathbf{N}\boldsymbol{n}'+\boldsymbol{n}}$$

$$= \sum_{\boldsymbol{n}\in\left\{\begin{smallmatrix}\text{mod-}\mathbf{N}\\ \text{columns}\end{smallmatrix}\right\}} \mathrm{e}^{-\mathrm{j}2\pi \boldsymbol{k}\mathbf{N}^{-1}\boldsymbol{n}} \left[\sum_{\boldsymbol{n}'} g_{\mathbf{N}\boldsymbol{n}'+\boldsymbol{n}}\, s_{\mathbf{N}\boldsymbol{n}'+\boldsymbol{n}}\right]. \tag{30}$$

While the left side of (30) is indexed on $\boldsymbol{k}''= \boldsymbol{k}'\mathbf{N}+\boldsymbol{k}$, it is not actually dependent on $\boldsymbol{k}'$ in any way, so $X_{\boldsymbol{k}''}$ is not changed by adding an offset of form $\boldsymbol{k}_\Delta\mathbf{N}$ to $\boldsymbol{k}''$, where $\boldsymbol{k}_\Delta \in \mathbb{Z}^2$ is arbitrary. This simply reflects that the array factor is periodic in the steering vector. We can therefore use these two decompositions for $\boldsymbol{k}''$and an additional relationship from (16)

$$\boldsymbol{k}''= \boldsymbol{k}'\mathbf{N}+\boldsymbol{k}$$
$$\boldsymbol{k}''= \boldsymbol{k}'_{\text{mod}}\mathbf{N}+\boldsymbol{k}'' \bmod \mathbf{N}$$
$$\boldsymbol{k}'' \bmod \mathbf{N} = \boldsymbol{k} \bmod \mathbf{N},$$

to assert that

$$\underset{\substack{\text{used in}\uparrow\\ \text{system design}}}{X_{\boldsymbol{k}''}} = X_{\underbrace{\boldsymbol{k}''\bmod\mathbf{N}}_{\substack{\text{used only to}\\ \text{equate the others}}}} = X_{\underbrace{\boldsymbol{k}\bmod\mathbf{N}}} = \underset{\substack{\uparrow\text{used in}\\ \text{FFT design}}}{X_{\boldsymbol{k}}}.$$

We'll use the left form for array system design in Section 3, the right form for FFT design in Section 4, and the middle two as a convenient way to match up those two beam-indexing systems. In other words, the beam output we need and the beam output we are computing to meet that need must have the same index modulo $\mathbf{N}$.

With this understood then, we can write the general result as

$$X_{\boldsymbol{k}\bmod\mathbf{N}} = \sum_{\boldsymbol{n}\in\left\{\begin{smallmatrix}\text{mod-}\mathbf{N}\\ \text{columns}\end{smallmatrix}\right\}} x_{\boldsymbol{n}}\, \mathrm{e}^{-\mathrm{j}2\pi \boldsymbol{k}\mathbf{N}^{-1}\boldsymbol{n}}, \tag{31}$$

$$x_{\boldsymbol{n}} \triangleq \sum_{\boldsymbol{n}'} g_{\mathbf{N}\boldsymbol{n}'+\boldsymbol{n}}\, s_{\mathbf{N}\boldsymbol{n}'+\boldsymbol{n}}, \tag{32}$$

revealing the beam output as a function of beam index $\boldsymbol{k} \bmod \mathbf{N}$ to be a generalized DFT of $|\mathbf{N}|$ *weighted and folded element outputs* $\left\{x_{\boldsymbol{n}} : \boldsymbol{n} \in \left\{\begin{smallmatrix}\text{mod-}\mathbf{N}\\ \text{columns}\end{smallmatrix}\right\}\right\}$, where the folding sum has combined those weighted element outputs that have the same indices mod $\mathbf{N}$. It is this folding in (32) that frees us from any requirement that the size of the array match the size of the transform.

	unnormalized	*normalized*			*beam index*	
			array plane			
		unit vector	*visible region*	*whole plane*	*gross*	*DFT*
wave DOA	$\boldsymbol{\kappa}$	$\widehat{\boldsymbol{\kappa}}$	$\underline{\widehat{\boldsymbol{\kappa}}}$	$\underline{\boldsymbol{\kappa}}$		
steering DOA	$\boldsymbol{\kappa}_s$	$\widehat{\mathbf{k}}''$	$\underline{\widehat{\mathbf{k}}}''$	$\underline{\mathbf{k}}''$	$\boldsymbol{k}''$	$\boldsymbol{k}$

Table 1: The frightening menagerie of notations relating to DOA (including some yet to be introduced in the text).

Periodicity of the weighted and folded element outputs. If we were to remove the restriction of $\boldsymbol{n}$ to $\left\{\begin{smallmatrix}\text{mod-N}\\ \text{columns}\end{smallmatrix}\right\}$ in (32), the weighted and folded element outputs $x_{\boldsymbol{n}}$ would in fact be well defined for all $\boldsymbol{n} \in \mathbb{Z}^2$. No information is lost in restricting $\boldsymbol{n}$ to $\left\{\begin{smallmatrix}\text{mod-N}\\ \text{columns}\end{smallmatrix}\right\}$ however, because the easy-to-demonstrate periodicity of $x_{\boldsymbol{n}}$ in $\boldsymbol{n}$ makes other values redundant.

3 System-design examples

This section develops the basics of a design discipline for a DFT-steered array by working through a series of examples, beginning with classic ones.

The many DOA notations used in the discussion to follow are somewhat bewildering, so they are summarized for convenience in Table 1.

A basis for direction cosines. Heretofore the array plane has been the span of the basis-vector columns of $\mathbf{B}$, but these in general are not orthogonal. Of course the rows of dual basis matrix $\mathbf{B}^+ = (\mathbf{B}^T\mathbf{B})^{-1}\mathbf{B}^T$ are linearly independent combinations of the basis-vector columns of basis matrix $\mathbf{B}$, so the rows of $\mathbf{B}^+$ also span the array plane and could be taken as a basis for it, but those rows may not be orthogonal either. As it is convenient to have an alternative, orthogonal array-plane basis for plotting, sketching, and discussion, let the first and second columns respectively of 3×2 *direction-cosine basis matrix* $\mathbf{P}$ be orthogonal array-plane unit vectors pointing in the azimuth and elevation directions normal to the boresight direction. If we assume the array is in a vertical plane and that we are at the array and looking outward and generally towards the horizon along the boresight direction, these azimuth and elevation directions are to the right and upward. We will generally plot the location of an arbitrary three-vector array-plane row vector $\underline{\boldsymbol{\kappa}}_{\text{arb}}$ using its $(\text{azimuth}, \text{elevation})$ pair of *generalized direction cosines* $\underline{\boldsymbol{\kappa}}_{\text{arb}}\mathbf{P}$. These are the traditional direction cosines of course when the sum of their squares is upper bounded by unity, as it is for $\underline{\widehat{\boldsymbol{\kappa}}}\mathbf{P}$ for example.

3.1 The Classic $\lambda/2$ square-grid design

Array-Factor Plots. The Fig. 2 example plot of the magnitude of an array factor illustrates plot conventions and key array-factor features.

> *Plots of array factors in this paper do not "go with" particular DFT-beamsteering designs in any aspect other than large-scale array-factor periodicity. Other array-factor features, including beamwidth, can be designed independently of steering considerations, The array factors shown here are therefore only for general illustration.*

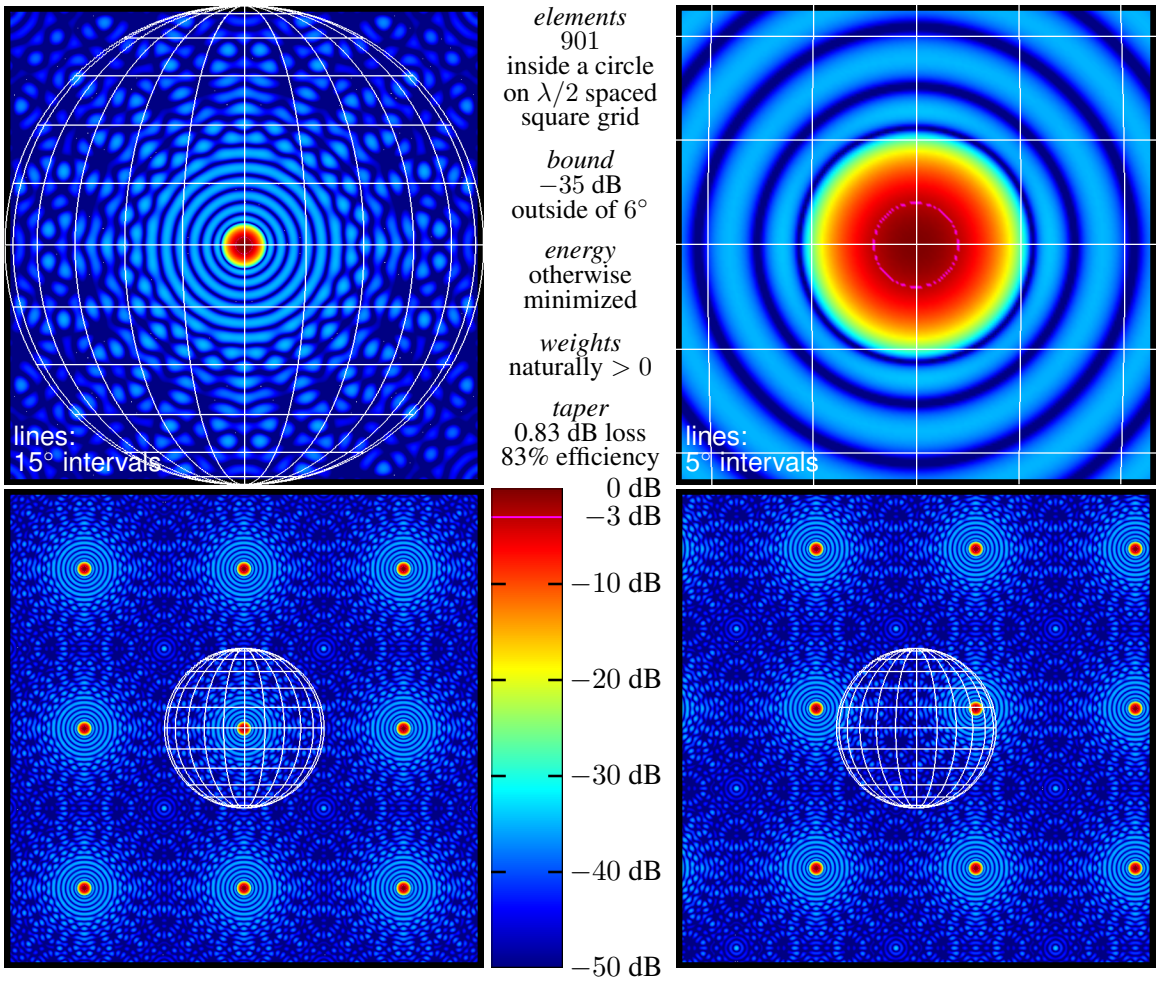


Figure 2: Example circular-beam array factor for a $\lambda/2$-spaced square-grid array, plotted as color-coded magnitude versus the generalized direction cosines $\underline{\boldsymbol{\kappa}}\mathbf{P}$ of array-plane vector $\underline{\boldsymbol{\kappa}}$ that becomes normalized array-plane DOA wave vector $\underline{\widehat{\boldsymbol{\kappa}}}$ inside the visible region represented here by a unit-radius world map. Lower right: example of nonzero steering. Other: unsteered, at three zoom levels.

(They were optimized using second-order cone programming, taking advantage of the symmetry-based simplifications discussed in the Allerton paper [4] and using the software discussed in [7].)

In Fig. 2 the decibel array-factor magnitude $20\log_{10}|G((\underline{\boldsymbol{\kappa}} - \underline{\widehat{\mathbf{k}}}'')\mathbf{B})|$ is represented by color and is plotted versus the generalized direction cosines $\underline{\boldsymbol{\kappa}}\mathbf{P}$ of normalized array-plane vector $\underline{\boldsymbol{\kappa}}$ with the origin at the center of the unit-radius "world map."

The circular $\|\underline{\boldsymbol{\kappa}}\| < 1$ interior of that world represents what is traditionally termed the *visible region* of the array plane. Inside the visible region $\underline{\boldsymbol{\kappa}}$ can be taken as the projection $\underline{\widehat{\boldsymbol{\kappa}}}$ of a normalization $\widehat{\boldsymbol{\kappa}} = \lambda\boldsymbol{\kappa}$ of physical DOA wave vector $\boldsymbol{\kappa}$, and it is the array factor at these $\underline{\widehat{\boldsymbol{\kappa}}}$ values that governs the array's response to signals. Inside the visible region the Cartesian plot position gives the traditional direction cosines $\underline{\widehat{\boldsymbol{\kappa}}}\mathbf{P}$ corresponding to the unit vector $\widehat{\boldsymbol{\kappa}}$ with actual azimuth and elevation angles given respectively by what appears on the world map to be east longitude, relative

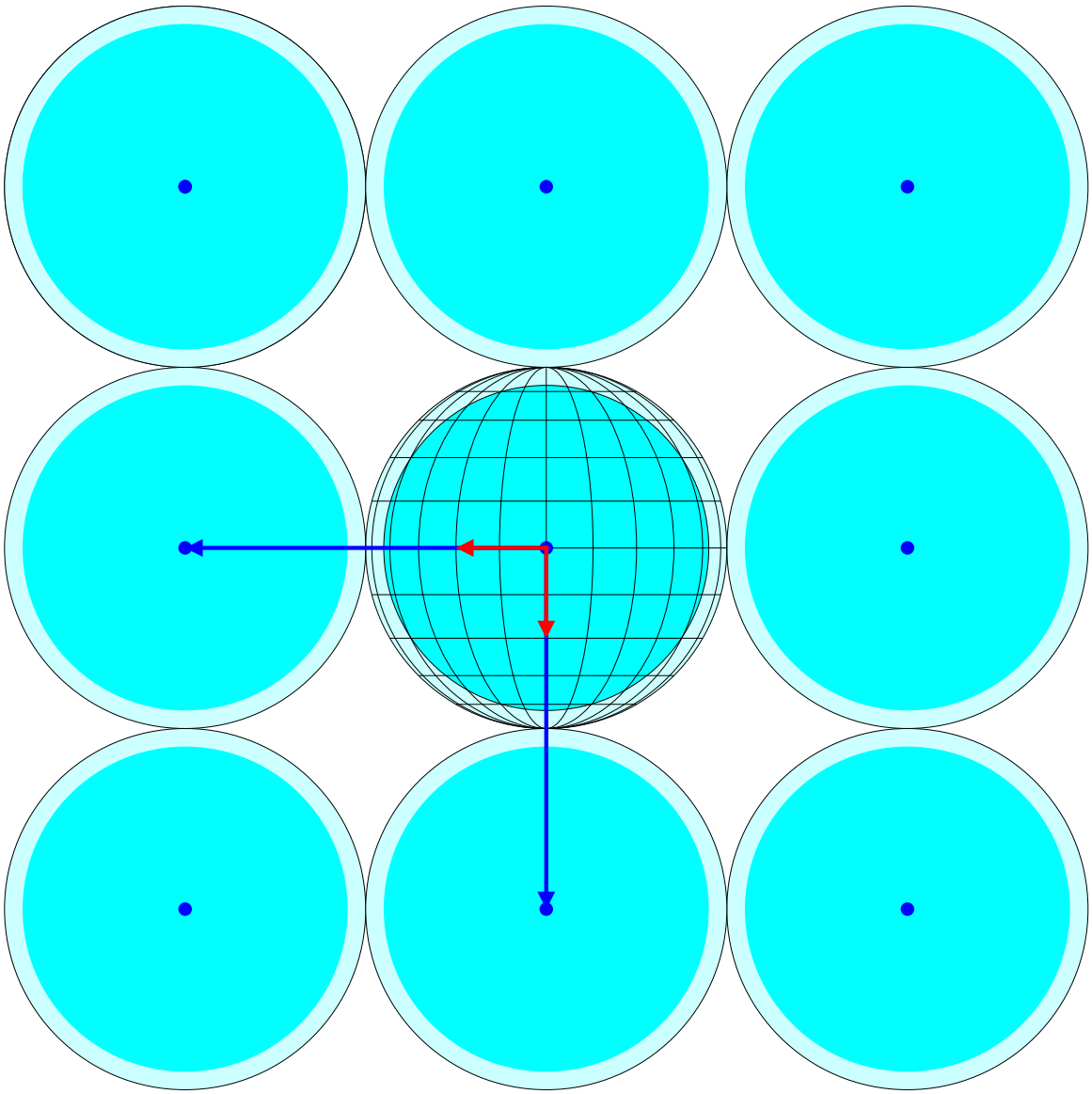


Figure 3: This is the classic Nyquist design on a square grid. The rows of dual basis matrix $\mathbf{B}^+$, plotted here as long vectors using generalized direction cosines $\mathbf{B}^+\mathbf{P}$, determine array-factor periodicity. The columns of basis matrix $\mathbf{B}$ are plotted here as short vectors using generalized direction cosines $\mathbf{P}^T\mathbf{B}$ and are scaled by λ to determine unnormalized element-location lattice $\lambda\mathbf{B}\mathbb{Z}^2$. Here the columns of $\mathbf{B}$ are orthogonal with length $1/2$ and so indicate $\lambda/2$ element spacing on a square grid.

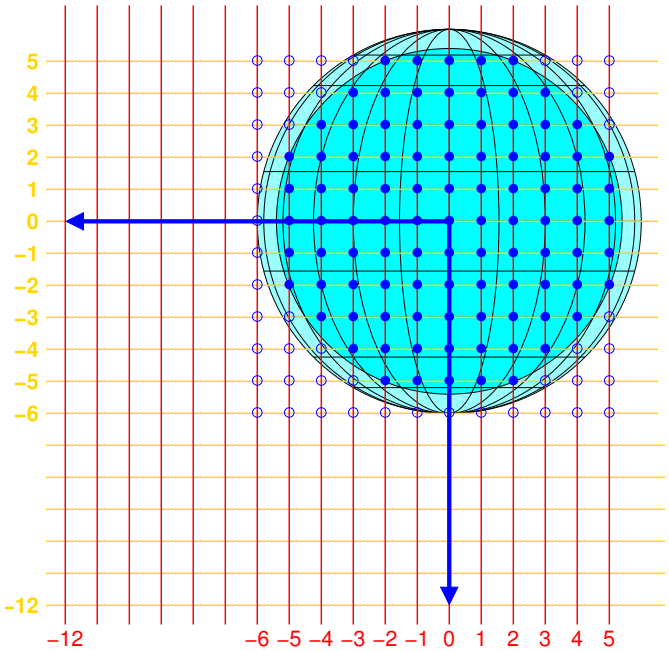


Figure 4: The choices for steering vector $\underline{\mathbf{k}}''$, the steering lattice, have generalized direction cosines $\underline{\mathbf{k}}''\mathbf{P}$ marked here by grid-line intersections ┼. That lattice must contain the basis-vector rows of dual basis matrix $\mathbf{B}^+$ plotted here as generalized direction cosines $\mathbf{B}^+\mathbf{P}$. Coordinates $\boldsymbol{k}'' = [\,|\,, —\,]$ of each such vector in terms of the | and — grid lines determine steering density matrix $\mathbf{N}$. An array-factor period's worth of the steering lattice, those with beam-index vectors $\boldsymbol{k}''$in $\left[\mathbb{Z}^2/\mathbb{Z}^2\mathbf{N}\right]$, are marked with **dots** (solid or hollow). FFT outputs need not be implemented where those steering vectors fall outside (hollow **dots**) the steering region.

---

to the vertical meridian, and north latitude.

Planar steering vector $\widehat{\underline{\mathbf{k}}}''$ is a parameter and is zero in Fig. 2 except for the plot on the lower right, which illustrates that steering amounts to simple translation in $\underline{\boldsymbol{\kappa}}$. Plotting versus azimuth and elevation angles instead would have destroyed this elegant translation property.

Steered array factor $G((\underline{\boldsymbol{\kappa}} - \widehat{\underline{\mathbf{k}}}'')\mathbf{B})$ governs signal response for visible $\underline{\boldsymbol{\kappa}}$ but is certainly well-defined for arbitrary $\underline{\boldsymbol{\kappa}}$. Further, this array factor is periodic in $\underline{\boldsymbol{\kappa}}$ because FIR response $G(\boldsymbol{f})$ in (24) is periodic in $\boldsymbol{f}$, the offsetting of which by any row two-vector of integers leaves $G(\boldsymbol{f})$ unchanged. In array factor $G((\underline{\boldsymbol{\kappa}} - \widehat{\underline{\mathbf{k}}}'')\mathbf{B})$ offsetting $\underline{\boldsymbol{\kappa}}$ by any element of $\mathbb{Z}^2\mathbf{B}^+$ offsets $\boldsymbol{f} = (\underline{\boldsymbol{\kappa}} - \widehat{\underline{\mathbf{k}}}'')\mathbf{B}$ by an integer two vector, because $\mathbf{B}^+\mathbf{B} = \mathbf{I}$. The unsteered array factor $G(\underline{\boldsymbol{\kappa}}\mathbf{B})$, for example, is for any $\underline{\boldsymbol{\kappa}} \in \mathbb{Z}^2\mathbf{B}^+$ the same as for $\underline{\boldsymbol{\kappa}} = 0$, the center of the boresight beam. The row-vector elements of *boresight replication lattice* $\mathbb{Z}^2\mathbf{B}^+$ are the centers of the periodically replicated beams in the lower-left plot in Fig. 2.

The array factor for invisible $\underline{\boldsymbol{\kappa}}$ is irrelevant to signal response, but its values at such $\underline{\boldsymbol{\kappa}}$ are very relevant to noise analysis, as when identical individual preamplifiers are used at each element to set the system noise figure, the front-end noise they produce results in array rms output noise proportional to the rms value of the array factor over an entire period—for example the single-period area covered by the upper-left plot in Fig. 2—and not just over the visible region. The invisible response therefore directly impacts SNR gain or, equivalently, aperture efficiency or taper loss, and for that reason the array-factor optimization should generally be configured to extend the sidelobe region outside the visible region and across the invisible region.

Grating-lobe prevention. Suppose the bright-blue part of the visible region in the Fig. 3 diagram is a *steering region* comprising all values of steering vector $\widehat{\underline{\mathbf{k}}}''$, all possible beam-center values, to be used in a particular system design. Of course with DFT beamsteering it would be a discrete set, but here it is approximated as a simply connected region for convenience. When steering places the beam center on this steering region's edge the beam's transition region, the rolloff of the array

factor to the sidelobe level, will extend outside the steering region. The additional area covered by that transition region as steering vector $\widehat{\underline{\mathbf{k}}}''$ traverses the steering region is sketched in pale blue and here will be termed the *steering penumbra*. The steering region and penumbra are duplicated periodically in $\underline{\boldsymbol{\kappa}}$ because the entire array factor is. If any of the invisible-region periodic replicas of the steering region with penumbra intrudes into the visible region at any point, then there is some allowable normalized array-plane steering vector $\widehat{\underline{\mathbf{k}}}''$ that will result in some part of the replica beam becoming visible, in which case that replica beam is termed a *grating lobe* (standard term). A replica beam that does not so intrude might be called a *potential grating lobe* (nonstandard term).

Many components of array cost are directly proportional to the density of elements in the array plane. The normalized element-location lattice $\mathbf{B}\not{\mathbb{Z}}^2$ has density

$$\frac{1}{\sqrt{\left|\mathbf{B}^T\mathbf{B}\right|}} = \sqrt{\left|\mathbf{B}^+(\mathbf{B}^+)^T\right|}, \tag{33}$$

per unit square, so the density of unnormalized element-location lattice $\lambda\mathbf{B}\not{\mathbb{Z}}^2$ is $1/\lambda^2$ of this value. Generally a system designer wishes to minimize this density, or equivalently, to maximize the density of replicated steering regions in the plane. The usual approach involves four steps.

1. System requirements for steering the beam determine the size and shape of the steering region within the "world" coordinate system.

2. The beam's rolloff characteristics—these can be estimated prior to actual beam design—determine the boundaries of the larger steering penumbra.

3. One chooses the two rows of $\mathbf{B}^+$, vectors in the array plane, so that replicas translated by the elements of $\mathbb{Z}^2\mathbf{B}^+$ will be packed as densely as possible in the plane without replica penumbras intruding into the visible region. Sometimes *a priori* constraints on grid geometry must be considered here.

4. Finally, the basis vectors for element positions $\mathbf{B}\not{\mathbb{Z}}^2$ are determined by solving definition $\mathbf{B}^+ = (\mathbf{B}^T\mathbf{B})^{-1}\mathbf{B}^T$ for $\mathbf{B}$ in terms of $\mathbf{B}^+$. To do this, and using notation $\mathbf{B}^{+T}$ for $(\mathbf{B}^+)^T$, first invert $\mathbf{B}^+\mathbf{B}^{+T} = \big((\mathbf{B}^T\mathbf{B})^{-1}\mathbf{B}^T\big)\big(\mathbf{B}(\mathbf{B}^T\mathbf{B})^{-1}\big) = (\mathbf{B}^T\mathbf{B})^{-1}$ to obtain $(\mathbf{B}^+\mathbf{B}^{+T})^{-1} = \mathbf{B}^T\mathbf{B}$. Then multiply the latter on the right by definition $\mathbf{B}^+ = (\mathbf{B}^T\mathbf{B})^{-1}\mathbf{B}^T$ to obtain $(\mathbf{B}^+\mathbf{B}^{+T})^{-1}\mathbf{B}^+ = \mathbf{B}^T\mathbf{B}\,(\mathbf{B}^T\mathbf{B})^{-1}\mathbf{B}^T = \mathbf{B}^T$. Transposing gives the desired result, $\mathbf{B} = \mathbf{B}^{+T}\big(\mathbf{B}^+\mathbf{B}^{+T}\big)^{-1}$, which is just $\mathbf{B} = \mathbf{B}^{+T+T}$. (This remarkably awkward notation becomes just $\mathbf{B} = \mathbf{B}^{++}$ once we agree, as is matlab's `pinv()` convention, that for wide matrices, taking the pseudoinverse begins and ends with transposition and takes the pseudoinverse of a tall matrix in between.)

The classic example is that of Fig. 3, where the steering region and penumbra just fill the visible region and where an *a priori* design constraint in favor of a square element-layout grid dictates that the rows of $\mathbf{B}^+$ be chosen to be orthogonal and of equal vector length so that the columns of $\mathbf{B}$ that govern element placement will be orthogonal and of equal length as well. Then the choice of $\mathbf{B}^+$ rows that will make boresight replication lattice $\mathbb{Z}^2\mathbf{B}^+$ as dense as possible are those shown in Fig. 3. Of course those row vectors could be reversed, individually negated, or rotated together without disturbing the density relationships. The particular choice made here, however, will lead shortly to convenient beam indexing.

Nyquist element spacing. In this example replicating the world map periodically would produce a universe with an infinite number of worlds optimally packed in the plane, touching but not overlapping and as dense in the plane as possible given the requirement that boresight replication lattice $\mathbb{Z}^2\mathbf{B}^+$ and element-location lattice $\mathbf{B}\mathbb{Z}^2$ be square grids. When such optimal packing of visible-region worlds obtains, the elements are said to be *Nyquist* spaced.

We can use our direction-cosine basis matrix $\mathbf{P}$ to go from the graphical image in Fig. 3 to a mathematical formulation and write

$$\mathbf{B}^+ = \begin{bmatrix} -2 & 0 \\ 0 & -2 \end{bmatrix} \mathbf{P}^T \qquad \text{and} \qquad \mathbf{B} = \mathbf{P} \begin{bmatrix} -\frac{1}{2} & 0 \\ 0 & -\frac{1}{2} \end{bmatrix}.$$

Here the columns of $\mathbf{B}$ that govern the placement of elements on points of $\lambda\mathbf{B}\mathbb{Z}^2$ correspond to the classic Nyquist $\lambda/2$ nearest-neighbor spacing on a square grid.

Nyquist spacing is a useful reference point, but it is optimal only in the special case in which the steering region and penumbra together exactly fill the visible-region world, as in this example.

The classic steering lattice for the classic array. Above we removed the confinement of spatial frequency $\widehat{\underline{\boldsymbol{\kappa}}}$ to the visible region and examined the behavior of array factor $G((\underline{\boldsymbol{\kappa}} - \widehat{\underline{\mathbf{k}}}'')\mathbf{B})$ as a function of the unrestricted spatial frequency $\underline{\boldsymbol{\kappa}}$. Though we may not find it useful to steer outside the visible region, the mathematics and any associated realization will certainly not object to us doing so, so we can likewise remove the confinement of steering vector $\widehat{\underline{\mathbf{k}}}''$ to the visible region and consider steering beam sum (31) with an unrestricted steering vector $\underline{\mathbf{k}}''$ obtained by removing the visible-region restriction from (27) to obtain

$$\underline{\mathbf{k}}'' = -\boldsymbol{k}''\mathbf{N}^{-1}\mathbf{B}^+. \tag{34}$$

Then of course (34) steers array factor $G((\underline{\boldsymbol{\kappa}} - \underline{\mathbf{k}}'')\mathbf{B})$ as well.

A valuable alternative interpretation of (34) is that steering lattice $\mathbb{Z}^2\mathbf{N}^{-1}\mathbf{B}^+$ is a superlattice of boresight replication lattice $\mathbb{Z}^2\mathbf{B}^+$. This is so because an arbitrary element $\boldsymbol{k}_{\text{arb}}\mathbf{B}^+$ of $\mathbb{Z}^2\mathbf{B}^+$ is equal to $(\boldsymbol{k}_{\text{arb}}\mathbf{N})\mathbf{N}^{-1}\mathbf{B}^+$ and is thus a member of $\mathbb{Z}^2\mathbf{N}^{-1}\mathbf{B}^+$ also. To choose steering density matrix $\mathbf{N}$, one must therefore

1. choose a steering lattice that is a superlattice of the boresight replication lattice,
2. choose two points of steering lattice $\mathbb{Z}^2\mathbf{N}^{-1}\mathbf{B}^+$ that can function as a basis for it,
3. taking those steering-lattice basis vectors to be the rows of steering-lattice basis matrix $\mathbf{N}^{-1}\mathbf{B}^+$, determine the beam index $\boldsymbol{k}''$ required to steer $\underline{\mathbf{k}}''$ to each basis vector row of $\mathbf{B}^+$ in turn using (34), and
4. solve for $\mathbf{N}$.

Let us examine the last step more closely. Suppose we have, as per step 3, chosen the beam indices $\boldsymbol{k}''$ that will steer the now unrestricted beam to each of $[1\ 0]\,\mathbf{B}^+$ and $[0\ 1]\,\mathbf{B}^+$, the row basis vectors of the dual basis matrix $\mathbf{B}^+$ for the boresight replication lattice $\mathbb{Z}^2\mathbf{B}^+$ that governs array-factor periodicity. By (34) we then have

$$\begin{aligned} [1\ 0]\mathbf{B}^+ &= -\boldsymbol{k}_1''\mathbf{N}^{-1}\mathbf{B}^+ \\ [0\ 1]\mathbf{B}^+ &= -\boldsymbol{k}_2''\mathbf{N}^{-1}\mathbf{B}^+, \end{aligned}$$

which of course is just the single matrix equation

$$\mathbf{B}^+ = -\begin{bmatrix} \boldsymbol{k}_1'' \\ \boldsymbol{k}_2'' \end{bmatrix} \mathbf{N}^{-1}\mathbf{B}^+.$$

Multiplying on the right by $\mathbf{BN}$ and using $\mathbf{B}^+\mathbf{B} = \mathbf{I}$ results in

$$\mathbf{N} = -\begin{bmatrix} \boldsymbol{k}_1'' \\ \boldsymbol{k}_2'' \end{bmatrix}, \tag{35}$$

so the required choices of $\boldsymbol{k}_1''$ and $\boldsymbol{k}_2''$ are just the rows of $-\mathbf{N}$. This relationship is more useful in form (35), however, because there it informs our choice of steering density matrix $\mathbf{N}$.

Now consider the Fig. 3 design in particular, and suppose we create a steering lattice with $12^2 = 144$ beams per array-factor period by assigning the first and second rows of dual basis matrix $\mathbf{B}^+$ the beam-index vectors $\begin{bmatrix} -12 & 0 \end{bmatrix}$ and $\begin{bmatrix} 0 & -12 \end{bmatrix}$ respectively, so that

$$\mathbf{N} = -\begin{bmatrix} [-12 & 0] \\ [\ \ 0 & -12] \end{bmatrix} = \begin{bmatrix} 12 & 0 \\ 0 & 12 \end{bmatrix}.$$

Here the elements of $\mathbf{N}$ are positive, but this is not required. We could replace this $\mathbf{N}$ matrix with its negative here with no harm done, as the sign change would in the end have been compensated by the associated sign reversals in the indices $\underline{\mathbf{k}}''$used to select beams in particular directions.

We determined $\mathbf{N}$ from the steering lattice $\mathbb{Z}^2\mathbf{N}^{-1}\mathbf{B}^+$of possible values for unrestricted steering vector $\underline{\mathbf{k}}''$, but of course either, once known, determines the other. The schema for the current example is shown in Fig. 4, where $\underline{\mathbf{k}}''$is selected using (34) and the index $\boldsymbol{k}'' = [\,|\,, \text{—}\,]$ determined by the | and — grid lines associated with a + grid-line intersection. The $|\mathbf{N}| = 12^2 = 144$ possible $\underline{\mathbf{k}}''$ values in a particular array-factor period are shown as **dots**, with the steering-region $\widehat{\underline{\mathbf{k}}}''$ subset shown solid and the rest shown hollow. The beam-index vectors $\boldsymbol{k}''$associated with all of the **dots** together are the elements of a particular choice for remainder set $\left[\mathbb{Z}^2/\mathbb{Z}^2\mathbf{N}\right]$ that would be natural for the system designer.

A bricklayer's grid for the classic array. As an alternative, we can give the same classic Fig. 3 array design a steering lattice in which unrestricted steering vectors $\underline{\mathbf{k}}''$are offset in alternate rows, in brick-wall fashion. We could have spaced steering vectors more closely in elevation than in azimuth in the previous example and did not, but here we can and will, simply to illustrate the approach.

We begin with a low-density preliminary steering lattice by assigning the first and second rows of dual basis matrix $\mathbf{B}^+$the beam-index vectors $\begin{bmatrix} 1 & -1 \end{bmatrix}$ and $\begin{bmatrix} -2 & -2 \end{bmatrix}$ respectively, so that

$$\mathbf{N}_{\text{prelim}} = -\begin{bmatrix} [\ \ 1 & -1] \\ [-2 & -2] \end{bmatrix} = \begin{bmatrix} -1 & 1 \\ 2 & 2 \end{bmatrix}.$$

This will place the unrestricted steering vectors $\underline{\mathbf{k}}''$on the intersections in Fig. 5 of the ╱ and ╲ lines and results in the least-dense steering lattice $\mathbb{Z}^2\mathbf{N}_{\text{prelim}}^{-1}\mathbf{B}^+$ that is a superlattice of boresight replication lattice $\mathbb{Z}^2\mathbf{B}^+$ while having a brick layout and $2:1$ horizontal:vertical spacing. Here $\mathbb{Z}^2\mathbf{N}_{\text{prelim}}^{-1}\mathbf{B}^+$has $|\mathbf{N}_{\text{prelim}}| = 4$ times the density of $\mathbb{Z}^2\mathbf{B}^+$.

To scale up the density in what will be an FFT-friendly way in Section 4, we then choose our final steering lattice $\mathbb{Z}^2\mathbf{N}^{-1}\mathbf{B}^+$to be a superlattice of preliminary steering lattice $\mathbb{Z}^2\mathbf{N}_{\text{prelim}}^{-1}\mathbf{B}^+$by scaling preliminary density matrix $\mathbf{N}_{\text{prelim}}$ on the right with a nonsingular 2×2 integer scaling matrix $\mathbf{N}_{\text{scale}}$. It follows that $\mathbf{N} = \mathbf{N}_{\text{prelim}}\mathbf{N}_{\text{scale}}$ and therefore that $\mathbf{N}^{-1} = \mathbf{N}_{\text{scale}}^{-1}\mathbf{N}_{\text{prelim}}^{-1}$, which gives us the superlattice chain

$$\mathbb{Z}^2\mathbf{N}^{-1}\mathbf{B}^+ = \mathbb{Z}^2\mathbf{N}_{\text{scale}}^{-1}\mathbf{N}_{\text{prelim}}^{-1}\mathbf{B}^+ \supset \mathbb{Z}^2\mathbf{N}_{\text{prelim}}^{-1}\mathbf{B}^+ \supset \mathbb{Z}^2\mathbf{B}^+.$$

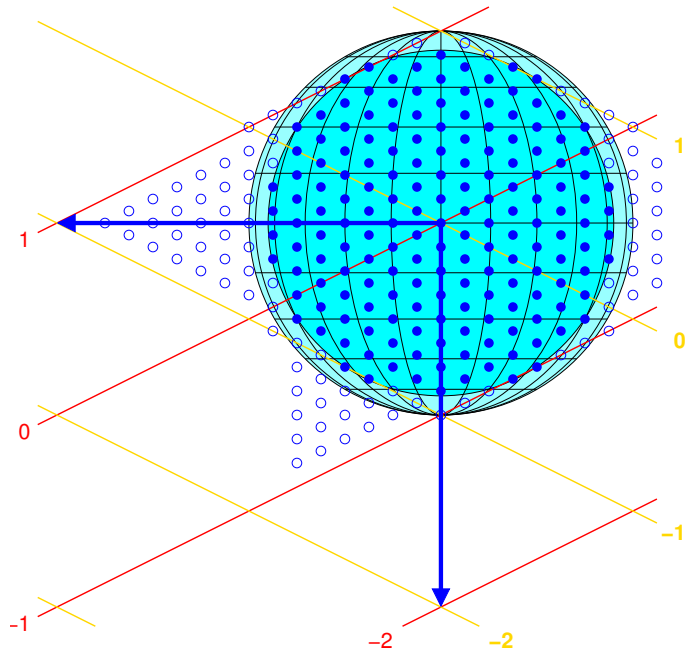


Figure 5: Here the same classic $\lambda/2$-spaced square element grid was used as in Fig. 4, along with the same steering region, but steering density matrix $\mathbf{N}$ was chosen to place steering vectors using the offset rows of a bricklayer's grid and with elevation spacing half of vertical spacing. The resulting steering-vector lattice is a slightly flattened version, by $\sqrt{3}/2$ vertical scaling actually, of a classic equilateral-triangle lattice.

Here we use the simplest possible choice of $\mathbf{N}_{\text{scale}}$, an identity matrix times some highly composite integer, say eight:

$$\mathbf{N} = \begin{bmatrix} -1 & 1 \\ 2 & 2 \end{bmatrix} \begin{bmatrix} 8 & 0 \\ 0 & 8 \end{bmatrix}.$$

Here matrix $8\mathbf{I}$ was used instead of a scalar eight in order to make explicit that $\mathbf{N}$ is in a form compatible with implementation of this system with the FFT developed in Section 4. This scaling by eight multiplies each grid label in Fig. 5 by eight as well. One period's worth of the resulting steering vectors are shown, with the period chosen for convenient figure sizing and obtained by starting with a large, horizontally extended, diamond-shaped period and cutting off and translating pieces by integer-vector left multiples of matrix $\mathbf{N}$ as convenient. There are $|\mathbf{N}| = 4 \times 8^2 = 256$ steering vectors in the period, of which the 169 associated with the steering region must be implemented.

3.2 The Classic $\lambda/\sqrt{3}$ triangular-grid design

Fig. 6 illustrates the basis geometry of the classic Nyquist design on an equilateral-triangular grid (generally termed a *hexagonal lattice* in mathematics [6]). This geometry gives the densest possible boresight replication lattice $\mathbb{Z}^2\mathbf{B}^+$ and so results in the lowest element density in the plane of any 2D Nyquist-spaced lattice array. By (33) that density is

$$\frac{1}{\lambda^2}\sqrt{\left|\begin{bmatrix} 4 & -2 \\ -2 & 4 \end{bmatrix}\right|} = \frac{\sqrt{12}}{\lambda^2}$$

or $\sqrt{3}/2 \approx 87\%$ of the $4/\lambda^2$ density of the $\lambda/2$-spaced square-grid array of the last section. Here

$$\mathbf{B}^+ = \begin{bmatrix} -\sqrt{3} & -1 \\ \sqrt{3} & -1 \end{bmatrix} \mathbf{P}^T \qquad \text{and} \qquad \mathbf{B} = \mathbf{P} \begin{bmatrix} -\frac{1/2}{\sqrt{3}} & \frac{1/2}{\sqrt{3}} \\ -\frac{1}{2} & -\frac{1}{2} \end{bmatrix}.$$

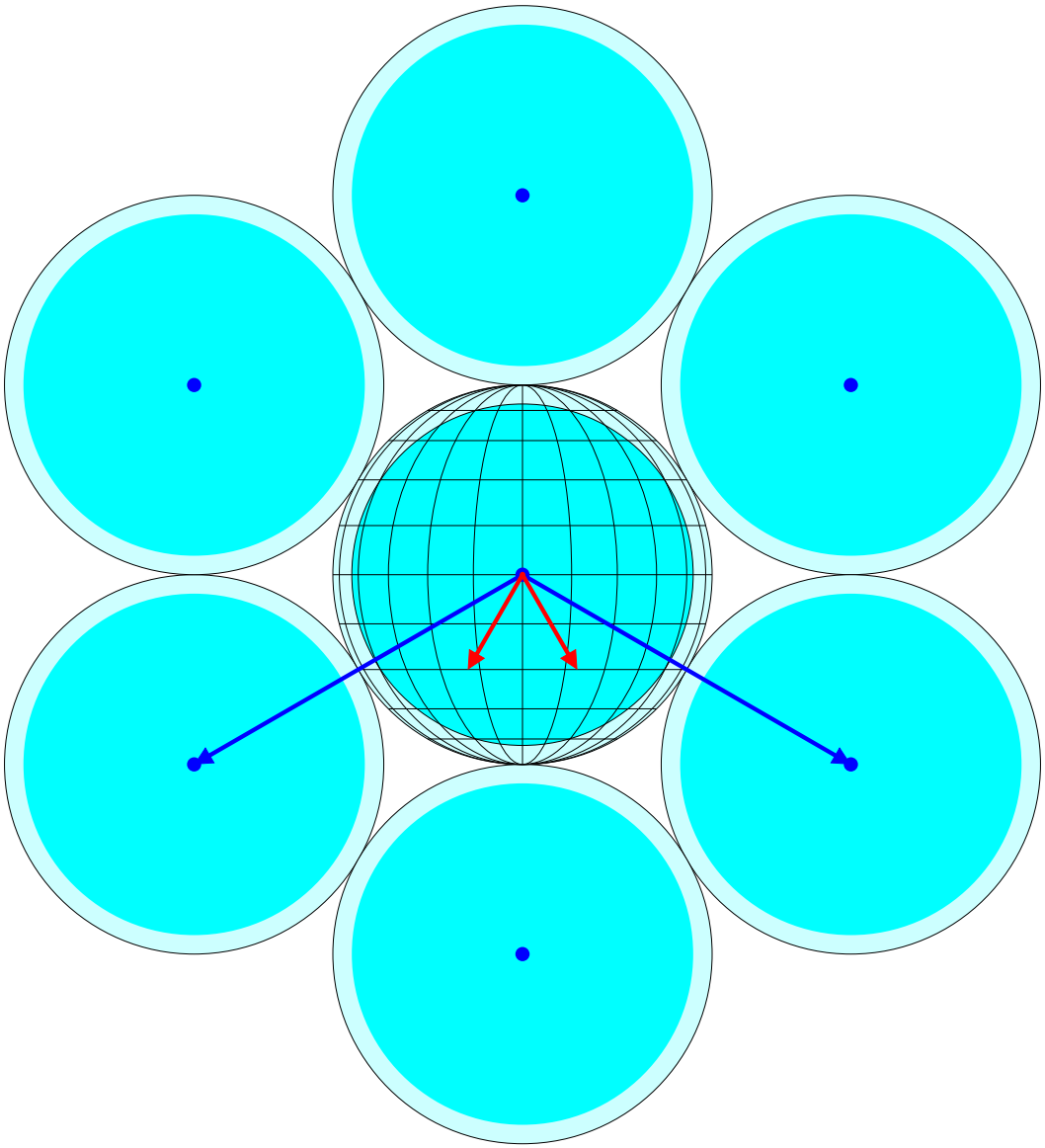


Figure 6: This is the classic Nyquist design on an equilateral-triangular grid. The rows of dual basis matrix $\mathbf{B}^+$, plotted here as long vectors using generalized direction cosines $\mathbf{B}^+\mathbf{P}$, determine array-factor periodicity. Here they are $120°$ apart and of length 2. The columns of basis matrix $\mathbf{B}$ are plotted here as short vectors using generalized direction cosines $\mathbf{P}^T\mathbf{B}$ and are scaled by λ to determine unnormalized element-location lattice $\lambda\mathbf{B}\mathbb{Z}^2$. Here those columns are $60°$ apart with length $1/\sqrt{3}$ and so indicate $\lambda/\sqrt{3}$ element spacing on an equilateral-triangular grid.

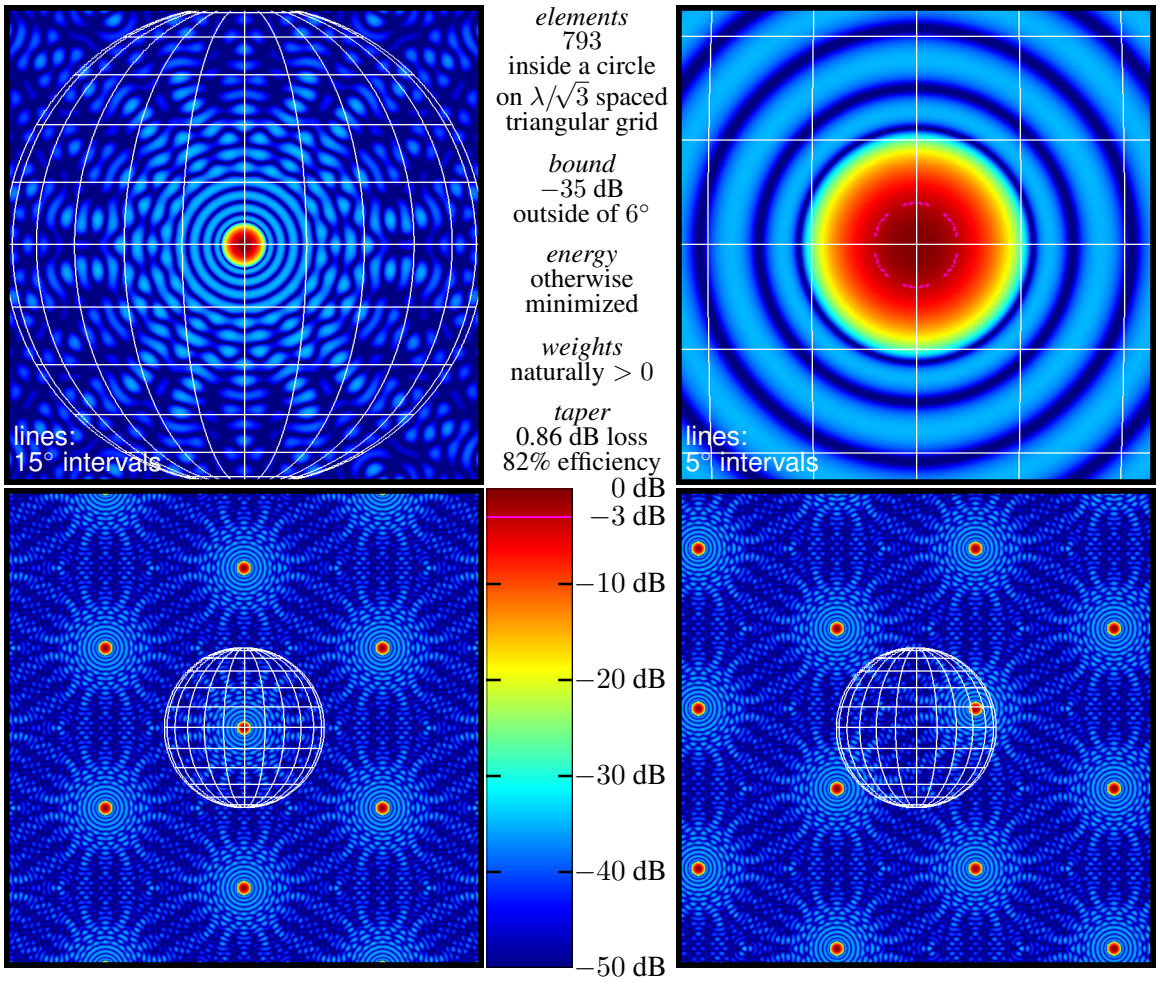


Figure 7: Example circular-beam array factor, for a $\lambda/\sqrt{3}$-spaced triangular-grid array, plotted as magnitude versus the generalized direction cosines $\underline{\boldsymbol{\kappa}}\mathbf{P}$ of array-plane vector $\underline{\boldsymbol{\kappa}}$ that becomes array-plane DOA wave vector $\underline{\widehat{\boldsymbol{\kappa}}}$ inside the world-map visible region. Lower right: example of nonzero steering. Other: unsteered, at three zoom levels.

Aside: For numerical-precision reasons it may sometimes be helpful to have radicals only in direction-cosine basis matrix $\mathbf{P}$ and in scalar scale factors. This can be done by choosing

$$\mathbf{P} = \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{6}} \\ -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{6}} \\ 0 & \frac{2}{\sqrt{6}} \end{bmatrix},$$

which puts boresight in the $[1\ 1\ 1]$ direction with

$$\mathbf{B}^{+} = \sqrt{\frac{2}{3}} \begin{bmatrix} -1 & 2 & -1 \\ 2 & -1 & -1 \end{bmatrix} \qquad \text{and} \qquad \mathbf{B} = \frac{1}{\sqrt{6}} \begin{bmatrix} 0 & 1 \\ 1 & 0 \\ -1 & -1 \end{bmatrix}.$$

An example array factor for this array geometry is shown in Fig. 7. The hexagonal-lattice character of the boresight replication lattice is immediately visible.

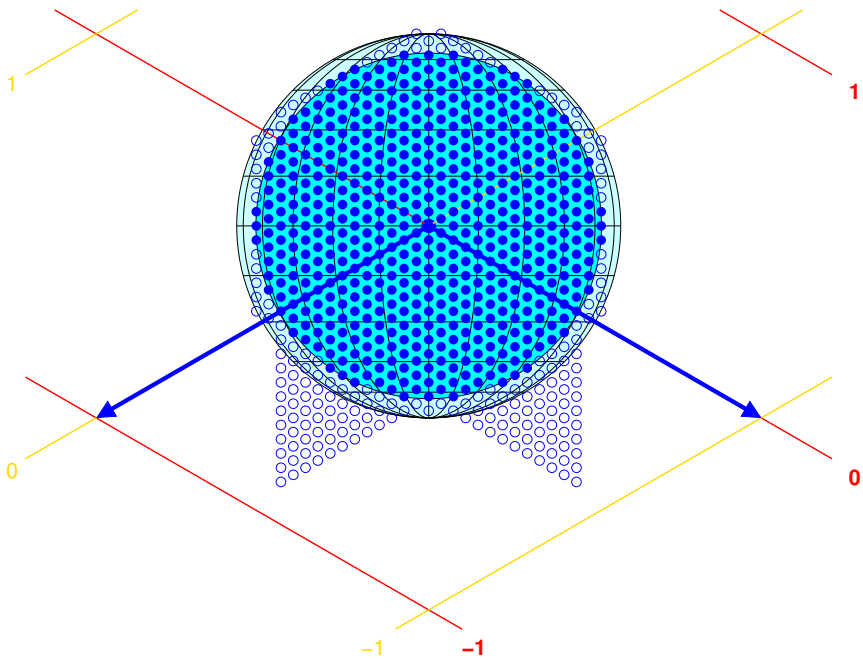


Figure 8: This example depicts the classic approach to DFT beam-steering for the Fig. 6 Nyquist design with $\lambda/\sqrt{3}$ element spacing on an equilateral-triangle grid. In this example each DFT period has $27^2 = 729$ beam-index vectors $\boldsymbol{k}''$, with each equal to $27 \times$ [,]. Only 547 of the associated steering vectors are inside the steering region.

The classic triangular steering lattice. The two-step design procedure at the end of Section 3.1 with

$$\mathbf{N} = -\begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} 27 & 0 \\ 0 & 27 \end{bmatrix}$$

results in the steering lattice illustrated in Fig. 8. The number 27 used here was arbitrary, of course, and in Section 4 we will certainly see that a power of three like this is compatible with FFT realization. Choosing $\mathbf{N}$ to be a scalar multiple of identity matrix $\mathbf{I}$, as was done here, results in a hexagonal steering lattice, because boresight replication lattice $\mathbb{Z}^2\mathbf{B}^+$ is hexagonal.

3.3 The general case: custom replication and steering lattices

In each example of Sections 3.1 and 3.2, a circularly shaped steering region led to a maximum amount of symmetry in boresight replication lattice $\mathbb{Z}^2\mathbf{B}^+$ and hence in element-location lattice $\mathbf{B}\mathbb{Z}^2$. But what if the steering region is more peculiarly shaped?

In Fig. 9 the array is to be steered from $-45°$ to $45°$ in azimuth and $-5°$ to $30°$ in elevation. Further, let us assume that grating lobes below about $-40°$ in elevation are harmless to system operation. This could be because the embedded pattern of the particular elements used strongly suppresses signals from that part of the visible region, or it might be due to some large and largely nonscattering obstruction in the array's field of view in those directions. (Is there such a thing?) Maybe it's for a combination of reasons. It's a system-design issue and doesn't matter here. Here let us simply suppose that it is so and see how we can take advantage of it to reduce the cost of the array. And let us also suppose that grating lobes are harmless at extreme vertical angles.

In Fig. 9 the replicated steering regions with penumbra are packed as tightly as possible consistent with grating-lobe requirements and consistent with giving replication offsets a lattice structure. The latter is important. For example, while lattices without vertical-midline reflection symmetry are certainly possible, here the vertical-midline reflection symmetry of the steering region prevents such a possibility from being useful. This is because the translation amounts come from boresight

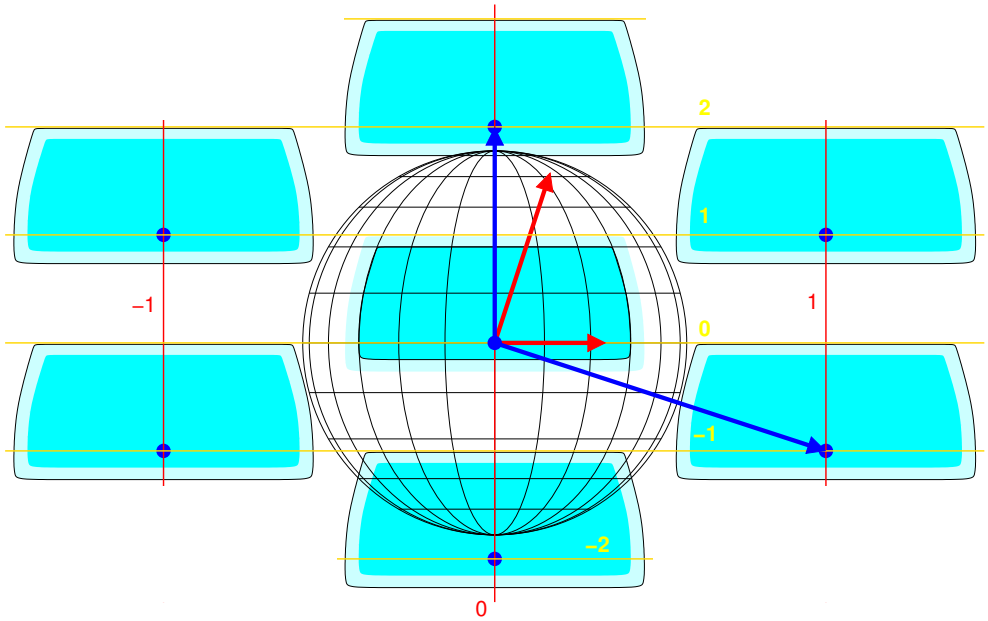


Figure 9: If grating lobes in some directions are acceptable, the less-dense element-location lattice $\mathbf{B}\mathbb{Z}^2$ resulting from a more-dense boresight replication lattice $\mathbb{Z}^2\mathbf{B}^+$ may lower costs without necessarily altering the range of steering-grid design alternatives.

---

replication lattice $\mathbb{Z}^2\mathbf{B}^+$, and every lattice is closed under negation. Lowering the left column of pictured replicas, for example, to bring the "left shoulder" replica's penumbra to the edge of the visible region would therefore require raising the "right hip" replica at the same time, immediately introducing grating lobes.

This boresight replication lattice $\mathbb{Z}^2\mathbf{B}^+$ has the pictured row basis vectors and was determined graphically after sliding steering regions with penumbra around in the software package used to make the drawing:

$$\mathbf{B}^+ = \frac{3}{80}\begin{bmatrix} 0 & 30 \\ 46 & -15 \end{bmatrix}\mathbf{P}^T.$$

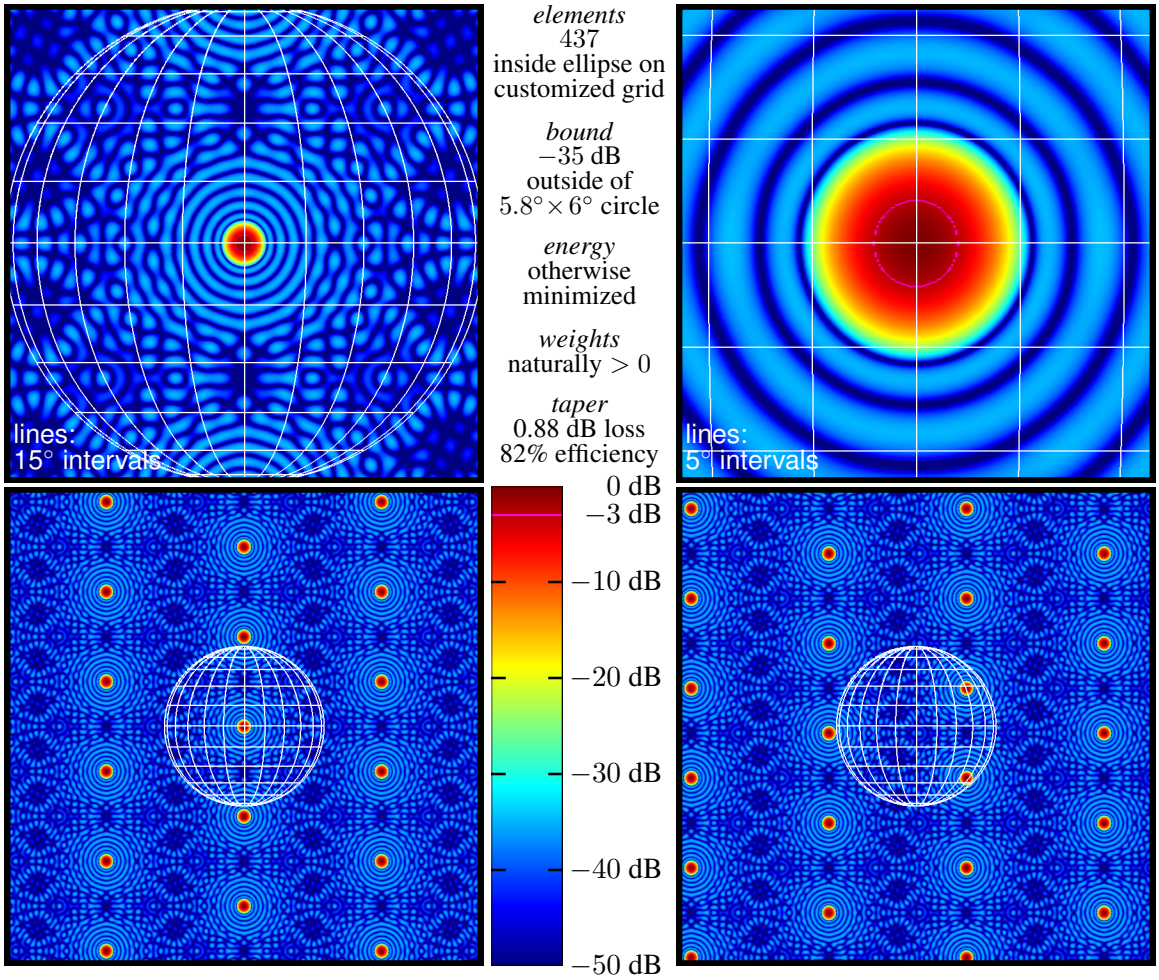
Basis $\mathbf{B} = \mathbf{B}^{+T+T}$ was then computed to be

$$\mathbf{B} = \mathbf{P}\,\frac{4}{207}\begin{bmatrix} 15 & 30 \\ 46 & 0 \end{bmatrix}. \tag{36}$$

Using the same procedure as in the previous examples, each box in Fig. 9 can be given n_{rows} rows by n_{cols} columns of beams using steering density matrix

$$\mathbf{N} = -\begin{bmatrix} 0 & 2 \\ 1 & -1 \end{bmatrix}\begin{bmatrix} n_{\text{cols}} & 0 \\ 0 & n_{\text{rows}} \end{bmatrix}.$$

We can see from the column basis vectors of $\mathbf{B}$ pictured in Fig. 9 that the normalized element-location lattice $\mathbf{B}\mathbb{Z}^2$ is more or less a vertically stretched version of the corresponding lattice in the equilateral-triangle-grid design of Section 3.2. The savings in element density here is substantial, as the density (33) of the unnormalized element-location lattice $\lambda\mathbf{B}\mathbb{Z}^2$ here is $\frac{621}{320}/\lambda^2 \approx 1.94/\lambda^2$, so there are just under two elements per square box with sides of length λ. Relative to the usual $\lambda/2$



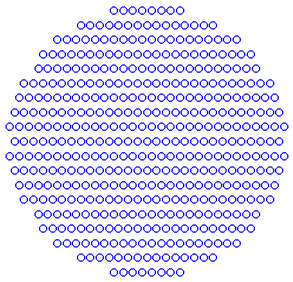
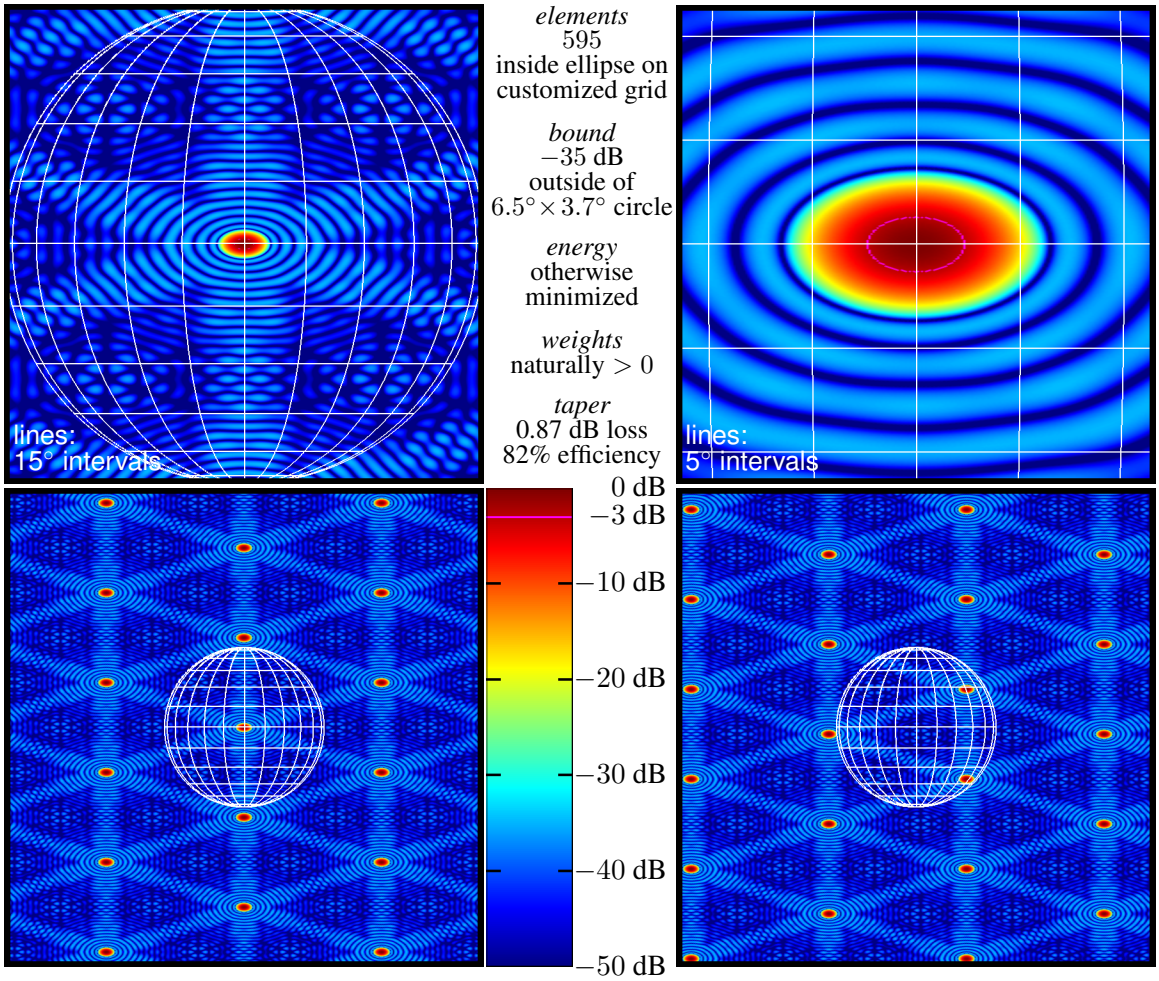


Figure 10: Example circular-beam array factor for a custom-grid array. Left: the element layout used, the perimeter of which is very slightly elliptical.

spacing on a square grid, the reduction is nearly 52%, and relative to $\lambda/\sqrt{3}$ spacing on a equilateral-triangular lattice the reduction is about 44%. We therefore expect reductions of those orders in the number of elements required to achieve a given beamshape and sidelobe level.

By way of example, a circular-beam array-factor design comparable to those of Figs. 2 and 7 but using the element layout specified by (36) is shown in Fig. 10. Though it was designed without looking back at the numbers of elements used in the square-grid and triangular-grid designs of Figs. 2 and 7, it turned out to require 437 elements, a reduction of about 51.5% and 45% respectively relative to those two earlier designs.

Each of the element-layout grids discussed, the square grid, the triangular grid, and this custom grid, fundamentally determines grating-lobe behavior but without limiting possible beam shapes.



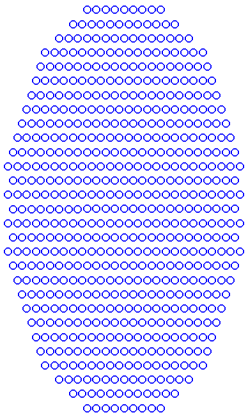


Figure 11: Above: example oval-beam array factor for a custom-grid array. Left: the element layout used.

This is illustrated by the Fig. 11 design, which uses the same custom grid as the design of Fig. 10 but aims instead for an oval beamshape. A significantly greater number of elements was used in order to obtain the narrower elevation beamwidth.

4 A 2D FFT with a matrix size parameter

The conventional FFT algorithm is generalized below, in Section 4.1, to allow the steering density matrix $\mathbf{N}$ to take up the position in the algorithm usually played by the number N of FFT points. Because of that connection $\mathbf{N}$ will here be termed the FFT *size matrix*. Particular forms of FFT size matrix and particular factorizations thereof then give the familiar special cases discussed in Section 4.2, which include the conventional radix-2 and mixed-radix FFT's, each in both decimation-in-time and decimation-in-frequency forms, and the 2D FFT in its most common form. Finally, the real utility of the general formulation is illustrated in Section 4.3, which develops an example custom FFT design targeted at the array steering system of the last, custom-grid example design of Section 3.3.

4.1 General derivation

The remaindered division of an integer vector by a square integer matrix discussed in Section 2.1 is at the heart of the derivation below, which begins with the factorization of FFT size matrix $\mathbf{N}$ and the subsequent decomposition of index column vector $\boldsymbol{n}$ into a multiple of one of the resulting factor matrices and a vector remainder. This amounts to a single vector digit's worth of radix decomposition of $\boldsymbol{n}$ using a matrix radix and allows the DFT expression to be factored. A similar radix decomposition of index row vector $\boldsymbol{k}$ but using the other factor of $\mathbf{N}$ then allows the factored DFT to be re-expressed using two smaller DFT expressions.

The discussion here assumes size parameter $\mathbf{N}$ is of size 2×2, but only for convenience of presentation and notation. The approach is general.

Factoring the size parameter. Suppose nonsingular 2×2 integer matrix $\mathbf{N}$, the FFT size, is composite in the sense that $\mathbf{N} = \mathbf{L}\mathbf{R}$, a product of two 2×2 integer matrices. What then becomes of $\left[\mathbb{Z}^2/\mathbf{N}\mathbb{Z}^2\right]$ and decomposition (1)? Can they be expressed in terms of $\mathbf{L}$ and $\mathbf{R}$?

In the spirit of (1) but using matrix $\mathbf{L}$ decompose element index $\boldsymbol{n}'' \in \mathbb{Z}^2$ as

$$\boldsymbol{n}'' = \mathbf{L}\boldsymbol{\mu} + \boldsymbol{\ell},$$

where $\boldsymbol{\ell} \in \left[\mathbb{Z}^2/\mathbf{L}\mathbb{Z}^2\right]$. Then further decompose $\boldsymbol{\mu} \in \mathbb{Z}^2$ as

$$\boldsymbol{\mu} = \mathbf{R}\boldsymbol{n}' + \boldsymbol{m},$$

where $\boldsymbol{m} \in \left[\mathbb{Z}^2/\mathbf{R}\mathbb{Z}^2\right]$. Together these give

$$\boldsymbol{n}'' = \mathbf{L}(\mathbf{R}\boldsymbol{n}' + \boldsymbol{m}) + \boldsymbol{\ell} = \mathbf{N}\boldsymbol{n}' + \mathbf{L}\boldsymbol{m} + \boldsymbol{\ell},$$

which will be just decomposition (1) with

$$\boldsymbol{n} = \mathbf{L}\boldsymbol{m} + \boldsymbol{\ell}$$

if set $\{\mathbf{L}\boldsymbol{m} + \boldsymbol{\ell} \colon \boldsymbol{m} \in \left[\mathbb{Z}^2/\mathbf{R}\mathbb{Z}^2\right], \boldsymbol{\ell} \in \left[\mathbb{Z}^2/\mathbf{L}\mathbb{Z}^2\right]\}$ qualifies as a remainder set $\left[\mathbb{Z}^2/\mathbf{N}\mathbb{Z}^2\right]$.

It does in fact qualify. First, $\left[\mathbb{Z}^2/\mathbf{L}\mathbb{Z}^2\right]$ and $\left[\mathbb{Z}^2/\mathbf{R}\mathbb{Z}^2\right]$ contain $|\mathbf{L}|$ and $|\mathbf{R}|$ elements respectively, so there are $|\mathbf{L}||\mathbf{R}| = |\mathbf{N}|$ possible choices of $(\boldsymbol{\ell}, \boldsymbol{m})$ pairs in the decompositions above. These pairs yield distinct sums $\mathbf{L}\boldsymbol{m} + \boldsymbol{\ell}$ because $\mathbf{L}\boldsymbol{m}_1 + \boldsymbol{\ell}_1 = \mathbf{L}\boldsymbol{m}_2 + \boldsymbol{\ell}_2$ implies $\boldsymbol{\ell}_1 - \boldsymbol{\ell}_2 = \mathbf{L}(\boldsymbol{m}_2 - m_1)$, which is disallowed by the definition of $\left[\mathbb{Z}^2/\mathbf{L}\mathbb{Z}^2\right]$ unless both sides are zero, proving by contradiction that set $\{\mathbf{L}\boldsymbol{m} + \boldsymbol{\ell} \colon \boldsymbol{m} \in \left[\mathbb{Z}^2/\mathbf{R}\mathbb{Z}^2\right], \boldsymbol{\ell} \in \left[\mathbb{Z}^2/\mathbf{L}\mathbb{Z}^2\right]\}$ contains $|\mathbf{N}|$ elements.

Second, no two sums $\mathbf{L}\boldsymbol{m}+\boldsymbol{\ell}$ differ by a right multiple of $\mathbf{N}$ because $(\mathbf{L}\boldsymbol{m}_1+\boldsymbol{\ell}_1)-(\mathbf{L}\boldsymbol{m}_2+\boldsymbol{\ell}_2)=\mathbf{N}\boldsymbol{\nu}=\mathbf{L}\mathbf{R}\boldsymbol{\nu}$ can be rewritten as $\mathbf{L}(\boldsymbol{m}_1-m_2-\mathbf{R}\boldsymbol{\nu})=\boldsymbol{\ell}_2-\boldsymbol{\ell}_1$, which is again disallowed by the definition of $[\mathbb{Z}^2/\mathbf{L}\mathbb{Z}^2]$ unless both sides are zero, so $\boldsymbol{\ell}_2=\boldsymbol{\ell}_1$ and $\boldsymbol{m}_1-m_2=\mathbf{R}\boldsymbol{\nu}$. But the latter is also disallowed by the definition of $[\mathbb{Z}^2/\mathbf{R}\mathbb{Z}^2]$ unless each of its sides is zero, so $\boldsymbol{m}_1=m_2$, and we have proved by contradiction that no two sums $\mathbf{L}\boldsymbol{m}+\boldsymbol{\ell}$ differ by a right multiple of $\mathbf{N}$.

While we could work with arbitrary remainder sets $[\mathbb{Z}^2/\mathbf{L}\mathbb{Z}^2]$ and $[\mathbb{Z}^2/\mathbf{R}\mathbb{Z}^2]$, let us instead standardize them and take

$$[\mathbb{Z}^2/\mathbf{L}\mathbb{Z}^2]=\left\{\begin{smallmatrix}\text{mod-}\mathbf{L}\\ \text{columns}\end{smallmatrix}\right\}$$
$$[\mathbb{Z}^2/\mathbf{R}\mathbb{Z}^2]=\left\{\begin{smallmatrix}\text{mod-}\mathbf{R}\\ \text{columns}\end{smallmatrix}\right\}.$$

Having established that the defining requirements are met, we can now take

$$[\mathbb{Z}^2/\mathbf{N}\mathbb{Z}^2]=\{\mathbf{L}\boldsymbol{m}+\boldsymbol{\ell}\colon \boldsymbol{m}\in\left\{\begin{smallmatrix}\text{mod-}\mathbf{R}\\ \text{columns}\end{smallmatrix}\right\},\boldsymbol{\ell}\in\left\{\begin{smallmatrix}\text{mod-}\mathbf{L}\\ \text{columns}\end{smallmatrix}\right\}\}. \tag{37}$$

We will work both with this specific $[\mathbb{Z}^2/\mathbf{N}\mathbb{Z}^2]$ and with $\left\{\begin{smallmatrix}\text{mod-}\mathbf{N}\\ \text{columns}\end{smallmatrix}\right\}$. They are related through (17), which here takes the form

$$\left\{\begin{smallmatrix}\text{mod-}\mathbf{N}\\ \text{columns}\end{smallmatrix}\right\}=\{(\mathbf{L}\boldsymbol{m}+\boldsymbol{\ell}) \bmod \mathbf{N}\colon \boldsymbol{m}\in\left\{\begin{smallmatrix}\text{mod-}\mathbf{R}\\ \text{columns}\end{smallmatrix}\right\},\boldsymbol{\ell}\in\left\{\begin{smallmatrix}\text{mod-}\mathbf{L}\\ \text{columns}\end{smallmatrix}\right\}\} \tag{38}$$

and will soon become the relationship we'll use to split the FFT.

For row vectors we could standardize the remainder sets $[\mathbb{Z}^2/\mathbb{Z}^2\mathbf{L}]$ and $[\mathbb{Z}^2/\mathbb{Z}^2\mathbf{R}]$ but will not. The FFT designer can use $\left\{\begin{smallmatrix}\text{mod-}\mathbf{L}\\ \text{rows}\end{smallmatrix}\right\}$ and $\left\{\begin{smallmatrix}\text{mod-}\mathbf{R}\\ \text{rows}\end{smallmatrix}\right\}$ or choose any convenient alternative. Given particular choices for those remainder sets, we then use

$$[\mathbb{Z}^2/\mathbb{Z}^2\mathbf{N}]=\{\boldsymbol{s}\mathbf{R}+\boldsymbol{r}\colon \boldsymbol{s}\in[\mathbb{Z}^2/\mathbb{Z}^2\mathbf{L}],\boldsymbol{r}\in[\mathbb{Z}^2/\mathbb{Z}^2\mathbf{R}]\}, \tag{39}$$

and the decomposition for $\boldsymbol{k}\in[\mathbb{Z}^2/\mathbb{Z}^2\mathbf{N}]$ that it implies.

Splitting the DFT. Relationships

$$\boldsymbol{n}=(\mathbf{L}\boldsymbol{m}+\boldsymbol{\ell}) \bmod \mathbf{N} \tag{40}$$
$$\boldsymbol{n}=\mathbf{L}\boldsymbol{m}+\boldsymbol{\ell}-\mathbf{N}\boldsymbol{n}'_{\text{offset}} \tag{41}$$

are equivalent, with the latter following from the former for some $\boldsymbol{n}'_{\text{offset}}\in\mathbb{Z}^2$ by (15). Below we will use (40) in subscripts and (41) in superscripts, the latter to take advantage of

$$\begin{aligned}\mathrm{e}^{-\mathrm{j}2\pi\boldsymbol{k}\mathbf{N}^{-1}\boldsymbol{n}}&=\mathrm{e}^{-\mathrm{j}2\pi\boldsymbol{k}\mathbf{N}^{-1}(\mathbf{L}\boldsymbol{m}+\boldsymbol{\ell}-\mathbf{N}\boldsymbol{n}'_{\text{offset}})}\\&=\mathrm{e}^{-\mathrm{j}2\pi\boldsymbol{k}\mathbf{N}^{-1}(\mathbf{L}\boldsymbol{m}+\boldsymbol{\ell})}\mathrm{e}^{\mathrm{j}2\pi\boldsymbol{k}\boldsymbol{n}'_{\text{offset}}}\\&=\mathrm{e}^{-\mathrm{j}2\pi\boldsymbol{k}\mathbf{N}^{-1}(\mathbf{L}\boldsymbol{m}+\boldsymbol{\ell})},\end{aligned}$$

which of course makes the particular value taken by $\boldsymbol{n}'_{\text{offset}}$ irrelevant.

Using (40) and (41) then, the DFT in (31) becomes

$$\begin{aligned}X_{\boldsymbol{k}\bmod\mathbf{N}}&=\sum_{\boldsymbol{m}\in\left\{\begin{smallmatrix}\text{mod-}\mathbf{R}\\ \text{columns}\end{smallmatrix}\right\}}\sum_{\boldsymbol{\ell}\in\left\{\begin{smallmatrix}\text{mod-}\mathbf{L}\\ \text{columns}\end{smallmatrix}\right\}}x_{(\mathbf{L}\boldsymbol{m}+\boldsymbol{\ell})\bmod\mathbf{N}}\,\mathrm{e}^{-\mathrm{j}2\pi\boldsymbol{k}\mathbf{N}^{-1}(\mathbf{L}\boldsymbol{m}+\boldsymbol{\ell})}\\&=\sum_{\boldsymbol{\ell}\in\left\{\begin{smallmatrix}\text{mod-}\mathbf{L}\\ \text{columns}\end{smallmatrix}\right\}}\mathrm{e}^{-\mathrm{j}2\pi\boldsymbol{k}\mathbf{N}^{-1}\boldsymbol{\ell}}\sum_{\boldsymbol{m}\in\left\{\begin{smallmatrix}\text{mod-}\mathbf{R}\\ \text{columns}\end{smallmatrix}\right\}}x_{(\mathbf{L}\boldsymbol{m}+\boldsymbol{\ell})\bmod\mathbf{N}}\,\mathrm{e}^{-\mathrm{j}2\pi\boldsymbol{k}(\mathbf{L}\mathbf{R})^{-1}\mathbf{L}\boldsymbol{m}}\\&=\sum_{\boldsymbol{\ell}\in\left\{\begin{smallmatrix}\text{mod-}\mathbf{L}\\ \text{columns}\end{smallmatrix}\right\}}\mathrm{e}^{-\mathrm{j}2\pi\boldsymbol{k}\mathbf{N}^{-1}\boldsymbol{\ell}}\sum_{\boldsymbol{m}\in\left\{\begin{smallmatrix}\text{mod-}\mathbf{R}\\ \text{columns}\end{smallmatrix}\right\}}x_{(\mathbf{L}\boldsymbol{m}+\boldsymbol{\ell})\bmod\mathbf{N}}\,\mathrm{e}^{-\mathrm{j}2\pi\boldsymbol{k}\mathbf{R}^{-1}\boldsymbol{m}}.\end{aligned} \tag{42}$$

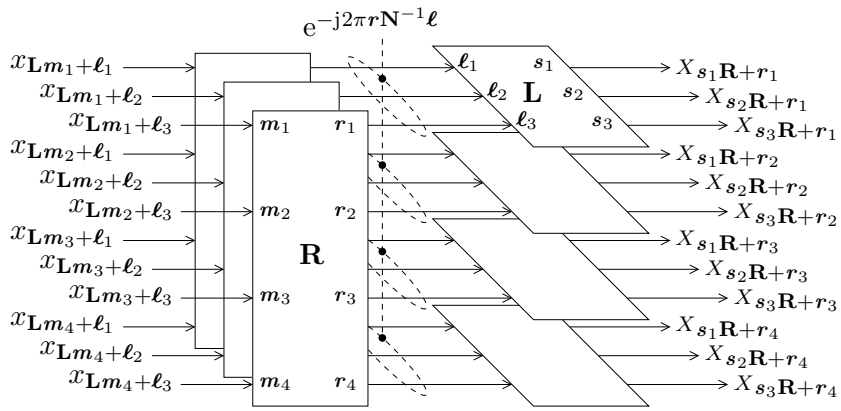


Figure 12: The structure (43) of a general FFT with size matrix $\mathbf{N}$, assuming $\mathbf{N} = \mathbf{LR}$ and using component FFT's with size matrices $\mathbf{L}$ and $\mathbf{R}$ and drawn here for $|\det(\mathbf{L})| = 3$ and $|\det(\mathbf{R})| = 4$. Twiddle factors $e^{-j2\pi r\mathbf{N}^{-1}\boldsymbol{\ell}}$ or, equivalently, $e^{-j2\pi(r\mathbf{R}^{-1})(\mathbf{L}^{-1}\boldsymbol{\ell})}$ are applied between the FFT steps.

The general form of the FFT. Decomposition $\boldsymbol{k} = \boldsymbol{s}\mathbf{R} + \boldsymbol{r}$ with $\boldsymbol{s} \in \left[\mathbb{Z}^2/\mathbb{Z}^2\mathbf{L}\right]$ and $\boldsymbol{r} \in \left[\mathbb{Z}^2/\mathbb{Z}^2\mathbf{R}\right]$ is implied by (39). Using it to rewrite (42) results in

$$\begin{aligned}
X_{(\boldsymbol{s}\mathbf{R}+\boldsymbol{r}) \bmod \mathbf{N}} &= \sum_{\boldsymbol{\ell}\in\left\{\substack{\text{mod-}\mathbf{L}\\ \text{columns}}\right\}} e^{-j2\pi(\boldsymbol{s}\mathbf{R}+\boldsymbol{r})\mathbf{N}^{-1}\boldsymbol{\ell}} \sum_{\boldsymbol{m}\in\left\{\substack{\text{mod-}\mathbf{R}\\ \text{columns}}\right\}} x_{(\mathbf{L}\boldsymbol{m}+\boldsymbol{\ell}) \bmod \mathbf{N}}\, e^{-j2\pi(\boldsymbol{s}\mathbf{R}+\boldsymbol{r})\mathbf{R}^{-1}\boldsymbol{m}} \\
&= \sum_{\boldsymbol{\ell}\in\left\{\substack{\text{mod-}\mathbf{L}\\ \text{columns}}\right\}} e^{-j2\pi\boldsymbol{s}\mathbf{R}(\mathbf{LR})^{-1}\boldsymbol{\ell}}\, e^{-j2\pi\boldsymbol{r}\mathbf{N}^{-1}\boldsymbol{\ell}} \sum_{\boldsymbol{m}\in\left\{\substack{\text{mod-}\mathbf{R}\\ \text{columns}}\right\}} x_{(\mathbf{L}\boldsymbol{m}+\boldsymbol{\ell}) \bmod \mathbf{N}}\, e^{-j2\pi\boldsymbol{s}\boldsymbol{m}} e^{-j2\pi\boldsymbol{r}\mathbf{R}^{-1}\boldsymbol{m}} \\
&= \sum_{\boldsymbol{\ell}\in\left\{\substack{\text{mod-}\mathbf{L}\\ \text{columns}}\right\}} \left[e^{-j2\pi(\boldsymbol{r}\mathbf{R}^{-1})(\mathbf{L}^{-1}\boldsymbol{\ell})} \sum_{\boldsymbol{m}\in\left\{\substack{\text{mod-}\mathbf{R}\\ \text{columns}}\right\}} x_{(\mathbf{L}\boldsymbol{m}+\boldsymbol{\ell}) \bmod \mathbf{N}}\, e^{-j2\pi\boldsymbol{r}\mathbf{R}^{-1}\boldsymbol{m}} \right] e^{-j2\pi\boldsymbol{s}\mathbf{L}^{-1}\boldsymbol{\ell}}
\end{aligned} \tag{43}$$

Some interpretation of (43) will reveal an FFT structure. The inner sum is a DFT of matrix size $\mathbf{R}$ of $x_{(\mathbf{L}\boldsymbol{m}+\boldsymbol{\ell}) \bmod \mathbf{N}}$ taken as a function of $\boldsymbol{m}$, with $\boldsymbol{\ell}$ treated as a parameter and where the "$\bmod \mathbf{N}$" is unnecessary, strictly speaking, because of the periodicity of weighted and folded element outputs $x_{\boldsymbol{n}}$ demonstrated at the end of Section 2.3. This inner-sum DFT is a function of $\boldsymbol{r}$ and $\boldsymbol{\ell}$ only and can be thought of as $|\det(\mathbf{L})|$ output sets of $|\det(\mathbf{R})|$ points each. After multiplication by so-called *twiddle factors* (standard name) $e^{-j2\pi(\boldsymbol{r}\mathbf{R}^{-1})(\mathbf{L}^{-1}\boldsymbol{\ell})}$ or, equivalently, $e^{-j2\pi\boldsymbol{r}\mathbf{N}^{-1}\boldsymbol{\ell}}$ these FFT output sets are taken collectively to be a function of $\boldsymbol{\ell}$ with $\boldsymbol{r}$ as a parameter and subjected in the outer sum to DFT's of matrix size $\mathbf{L}$, one for each value of parameter $\boldsymbol{r}$, resulting in $|\det(\mathbf{R})|$ output sets of $|\det(\mathbf{L})|$ points each.

The FFT of (43) is shown in block-diagram form in Fig. 12 for the specific case $|\det(\mathbf{L})| = 3$ and $|\det(\mathbf{R})| = 4$, which numbers then become the sizes of the component matrix-size FFT's shown as boxes. Here the optional "mod $\mathbf{N}$" steps on element (input) and beam (output) index vectors are omitted for readability, and

$$\begin{array}{lll}
\text{column vectors:} & \left[\mathbb{Z}^2/\mathbf{R}\mathbb{Z}^2\right] = \{\boldsymbol{m}_1, \boldsymbol{m}_2, \boldsymbol{m}_3, \boldsymbol{m}_4\} & \left[\mathbb{Z}^2/\mathbf{L}\mathbb{Z}^2\right] = \{\boldsymbol{\ell}_1, \boldsymbol{\ell}_2, \boldsymbol{\ell}_3\} \\
\text{row vectors:} & \left[\mathbb{Z}^2/\mathbb{Z}^2\mathbf{R}\right] = \{\boldsymbol{r}_1, \boldsymbol{r}_2, \boldsymbol{r}_3, \boldsymbol{r}_4\} & \left[\mathbb{Z}^2/\mathbb{Z}^2\mathbf{L}\right] = \{\boldsymbol{s}_1, \boldsymbol{s}_2, \boldsymbol{s}_3\}.
\end{array}$$

The component FFT's on the left are all identical, as are the component FFT's on the right. These

component FFT's can be implemented directly from DFT expression (31) or, if their size matrices are factorable, expressed as smaller FFT's using (43) recursively.

4.2 Familiar special cases

Special case: the ordinary 1D, radix-2 FFT. Suppose $\mathbf{N} = \left[\begin{smallmatrix} 2^K & 0 \\ 0 & 1 \end{smallmatrix}\right]$. Factorization $\mathbf{N} = \mathbf{LR} = \left[\begin{smallmatrix} 2 & 0 \\ 0 & 1 \end{smallmatrix}\right]\left[\begin{smallmatrix} 2^{K-1} & 0 \\ 0 & 1 \end{smallmatrix}\right]$ with

$$\begin{aligned}\left[\not{\mathbb{Z}}^2/\mathbf{R}\not{\mathbb{Z}}^2\right] &= \left[\mathbb{Z}^2/\mathbb{Z}^2\mathbf{R}\right]^T = \left\{\left[\begin{smallmatrix} 0 \\ 0 \end{smallmatrix}\right] \ldots \left[\begin{smallmatrix} 2^{K-1}-1 \\ 0 \end{smallmatrix}\right]\right\} \\ \left[\not{\mathbb{Z}}^2/\mathbf{L}\not{\mathbb{Z}}^2\right] &= \left[\mathbb{Z}^2/\mathbb{Z}^2\mathbf{L}\right]^T = \left\{\left[\begin{smallmatrix} 0 \\ 0 \end{smallmatrix}\right], \left[\begin{smallmatrix} 1 \\ 0 \end{smallmatrix}\right]\right\}\end{aligned}$$

gives the FFT of (43) and Fig. 12 an input stage that is a pair of 2^{K-1} point FFT's and an output stage comprising 2^{K-1} transforms of two inputs to two outputs, each of which, by (31), is the classic butterfly (standard name) computation, a sum and a difference. The twiddle factor shown in Fig. 12 here becomes

$$\mathrm{e}^{-\mathrm{j}2\pi \boldsymbol{r}\mathbf{N}^{-1}\boldsymbol{\ell}} = \mathrm{e}^{-\mathrm{j}2\pi \boldsymbol{r}\left[\begin{smallmatrix} 1/2^K & 0 \\ 0 & 1 \end{smallmatrix}\right]\boldsymbol{\ell}} = \mathrm{e}^{-\mathrm{j}2\pi r_1\ell_1/2^K + r_2\ell_2} = \mathrm{e}^{-\mathrm{j}2\pi r_1\ell_1/2^K}.$$

Recursively decomposing the input FFT's into smaller FFT's until the input FFT's are nothing but butterflies then yields the classic radix-2 decimation-in-time (in this beamsteering application, decimation-in-element-space) FFT.

Alternatively, let $\mathbf{N} = \mathbf{LR} = \left[\begin{smallmatrix} 2^{K-1} & 0 \\ 0 & 1 \end{smallmatrix}\right]\left[\begin{smallmatrix} 2 & 0 \\ 0 & 1 \end{smallmatrix}\right]$ with

$$\begin{aligned}\left[\not{\mathbb{Z}}^2/\mathbf{R}\not{\mathbb{Z}}^2\right] &= \left[\mathbb{Z}^2/\mathbb{Z}^2\mathbf{R}\right]^T = \left\{\left[\begin{smallmatrix} 0 \\ 0 \end{smallmatrix}\right], \left[\begin{smallmatrix} 1 \\ 0 \end{smallmatrix}\right]\right\} \\ \left[\not{\mathbb{Z}}^2/\mathbf{L}\not{\mathbb{Z}}^2\right] &= \left[\mathbb{Z}^2/\mathbb{Z}^2\mathbf{L}\right]^T = \left\{\left[\begin{smallmatrix} 0 \\ 0 \end{smallmatrix}\right] \ldots \left[\begin{smallmatrix} 2^{K-1}-1 \\ 0 \end{smallmatrix}\right]\right\}.\end{aligned}$$

The FFT of (43) and Fig. 12 then has an input stage comprising 2^{K-1}transforms of two inputs to two outputs, butterflies, and an output stage that is a pair of 2^{K-1}point FFT's. The twiddle factors are as before. Recursively decomposing the output FFT's into smaller FFT's until nothing but butterflies remain yields the classic radix-2 decimation-in-frequency (here decimation-in-beam-space) FFT.

Special case: an ordinary 1D, mixed-radix FFT. Suppose $\mathbf{N} = \left[\begin{smallmatrix} 3\times 2^K & 0 \\ 0 & 1 \end{smallmatrix}\right]$. Factorization $\mathbf{N} = \mathbf{LR} = \left[\begin{smallmatrix} 3 & 0 \\ 0 & 1 \end{smallmatrix}\right]\left[\begin{smallmatrix} 2^K & 0 \\ 0 & 1 \end{smallmatrix}\right]$ with

$$\begin{aligned}\left[\not{\mathbb{Z}}^2/\mathbf{R}\not{\mathbb{Z}}^2\right] &= \left[\mathbb{Z}^2/\mathbb{Z}^2\mathbf{R}\right]^T = \left\{\left[\begin{smallmatrix} 0 \\ 0 \end{smallmatrix}\right] \ldots \left[\begin{smallmatrix} 2^K-1 \\ 0 \end{smallmatrix}\right]\right\} \\ \left[\not{\mathbb{Z}}^2/\mathbf{L}\not{\mathbb{Z}}^2\right] &= \left[\mathbb{Z}^2/\mathbb{Z}^2\mathbf{L}\right]^T = \left\{\left[\begin{smallmatrix} 0 \\ 0 \end{smallmatrix}\right], \left[\begin{smallmatrix} 1 \\ 0 \end{smallmatrix}\right], \left[\begin{smallmatrix} 2 \\ 0 \end{smallmatrix}\right]\right\}\end{aligned}$$

gives the FFT of (43) and Fig. 12 specialized to the structure shown on the left in Fig. 13. The input stage comprises three 2^K point FFT's, and the output stage uses 2^K transforms of three inputs to three outputs, each of which, by (31), takes the form of the of three-winged dragonfly (nonstandard name) on the right in Fig. 13. The twiddle factor here becomes

$$\mathrm{e}^{-\mathrm{j}2\pi \boldsymbol{r}\mathbf{N}^{-1}\boldsymbol{\ell}} = \mathrm{e}^{-\mathrm{j}2\pi \boldsymbol{r}\left[\begin{smallmatrix} 1/(3\times 2^K) & 0 \\ 0 & 1 \end{smallmatrix}\right]\boldsymbol{\ell}} = \mathrm{e}^{-\mathrm{j}2\pi r_1\ell_1/(3\times 2^K) + r_2\ell_2} = \mathrm{e}^{-\mathrm{j}2\pi r_1\ell_1/(3\times 2^K)}.$$

The three input FFT's can be structured in classic radix-2 form as per the previous discussion.

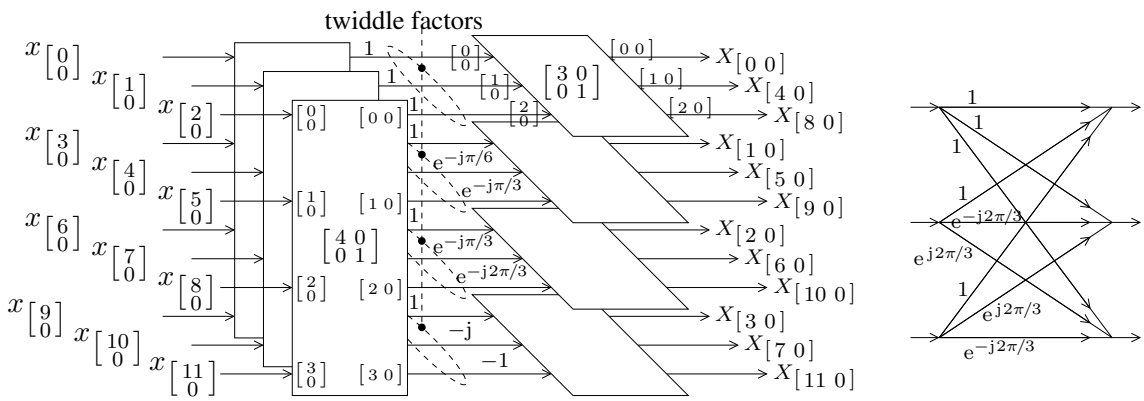


Figure 13: Left: This 12-point FFT, in effect in one dimension, was created from (43) with

$$\mathbf{L} = \left[\begin{smallmatrix}3&0\\0&1\end{smallmatrix}\right] \quad \text{and} \quad \mathbf{R} = \left[\begin{smallmatrix}4&0\\0&1\end{smallmatrix}\right]$$

and uses three four-point FFT's followed by four three-point DFT's. In terms of indices r_1 and ℓ_1, twiddle factors $\mathrm{e}^{-\mathrm{j}2\pi r_1\ell_1/12}$ evaluate as shown. Right: the three-point "dragonfly" DFT used four times on the left.

Alternatively, let $\mathbf{N} = \mathbf{LR} = \left[\begin{smallmatrix}2^K&0\\0&1\end{smallmatrix}\right]\left[\begin{smallmatrix}3&0\\0&1\end{smallmatrix}\right]$ with

$$\begin{aligned}\left[\mathbb{Z}^2/\mathbf{R}\mathbb{Z}^2\right] &= \left[\mathbb{Z}^2/\mathbb{Z}^2\mathbf{R}\right]^T = \left\{\left[\begin{smallmatrix}0\\0\end{smallmatrix}\right], \left[\begin{smallmatrix}1\\0\end{smallmatrix}\right], \left[\begin{smallmatrix}2\\0\end{smallmatrix}\right]\right\} \quad . \\ \left[\mathbb{Z}^2/\mathbf{L}\mathbb{Z}^2\right] &= \left[\mathbb{Z}^2/\mathbb{Z}^2\mathbf{L}\right]^T = \left\{\left[\begin{smallmatrix}0\\0\end{smallmatrix}\right] \ldots \left[\begin{smallmatrix}2^K-1\\0\end{smallmatrix}\right]\right\}\end{aligned}$$

The FFT of (43) and Fig. 12 then has an input stage comprising 2^K three-point DFT's, each a dragonfly mapping three inputs to three outputs, and an output stage that is three 2^K point FFT's. The twiddle factors are as before.

Special case: the ordinary 2D FFT using row and column FFT's. Suppose $\mathbf{N} = \left[\begin{smallmatrix}N_1&0\\0&N_2\end{smallmatrix}\right] = \mathbf{LR} = \left[\begin{smallmatrix}N_1&0\\0&1\end{smallmatrix}\right]\left[\begin{smallmatrix}1&0\\0&N_2\end{smallmatrix}\right]$.

$$\begin{aligned}\left[\mathbb{Z}^2/\mathbf{R}\mathbb{Z}^2\right] &= \left[\mathbb{Z}^2/\mathbb{Z}^2\mathbf{R}\right]^T = \left\{\left[\begin{smallmatrix}0\\0\end{smallmatrix}\right] \ldots \left[\begin{smallmatrix}0\\N_2-1\end{smallmatrix}\right]\right\} \\ \left[\mathbb{Z}^2/\mathbf{L}\mathbb{Z}^2\right] &= \left[\mathbb{Z}^2/\mathbb{Z}^2\mathbf{L}\right]^T = \left\{\left[\begin{smallmatrix}0\\0\end{smallmatrix}\right] \ldots \left[\begin{smallmatrix}N_1-1\\0\end{smallmatrix}\right]\right\}\end{aligned}$$

As illustrated in Fig. 14 for $N_1 = 3$ and $N_2 = 4$, this yields N_1 one-dimensional N_2-point FFT's on input rows followed by N_2 one-dimensional N_1-point FFT's to produce output columns. The twiddle factors $\mathrm{e}^{-\mathrm{j}2\pi \boldsymbol{r}\mathbf{N}^{-1}\boldsymbol{\ell}}$ in the center are equal to unity.

4.3 Custom FFT design

The Section 3.3 system design (page 27) featured $\mathbf{N} = \mathbf{LR} = \left[\begin{smallmatrix}0&-2\\-1&1\end{smallmatrix}\right]\left[\begin{smallmatrix}n_{\text{cols}}&0\\0&n_{\text{rows}}\end{smallmatrix}\right]$. Here let us develop the corresponding FFT structure.

We will need various remainder sets. In the middle of Section 2.1 we developed example remainder sets mod $\mathbf{N}$ geometrically using an example steering density matrix $\mathbf{N}$ that here is exactly our matrix $\mathbf{L}$, so (10) from that discussion now yields

$$\left\{\begin{smallmatrix}\text{mod-}\mathbf{L}\\\text{columns}\end{smallmatrix}\right\} = \left\{\left[\begin{smallmatrix}0\\0\end{smallmatrix}\right], \left[\begin{smallmatrix}-1\\0\end{smallmatrix}\right]\right\} \qquad \left\{\begin{smallmatrix}\text{mod-}\mathbf{L}\\\text{rows}\end{smallmatrix}\right\} = \left\{\left[\begin{smallmatrix}0&0\end{smallmatrix}\right], \left[\begin{smallmatrix}0&-1\end{smallmatrix}\right]\right\}.$$

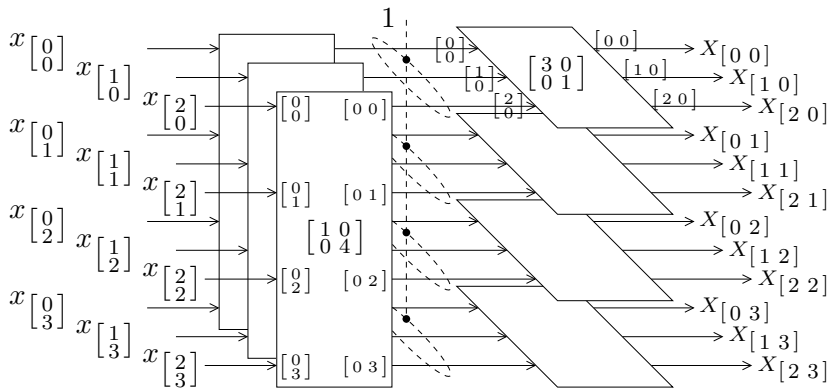


Figure 14: The structure of the FFT of (43) when

$$\mathbf{L} = \left[\begin{smallmatrix}3&0\\0&1\end{smallmatrix}\right] \quad \text{and} \quad \mathbf{R} = \left[\begin{smallmatrix}1&0\\0&4\end{smallmatrix}\right].$$

This choice of $\mathbf{L}$ and $\mathbf{R}$ yields three 1D FFT's on rows (the image is transposed relative to convention) followed by four 1D FFT's on columns. Swapping $\mathbf{L}$ and $\mathbf{R}$ would instead yield four 1D FFT's on columns followed by three 1D FFT's on rows and using unit twiddle factors in between.

If we then take $n_{\text{cols}} = 24$ and $n_{\text{rows}} = 12$ for the sake of specificity so that $\mathbf{R} = \left[\begin{smallmatrix}24&0\\0&12\end{smallmatrix}\right]$, we can use the straightforward graphical approach of Fig. 15 to arrive at

$$\begin{aligned}\left[\mathbb{Z}^2/\mathbf{R}\mathbb{Z}^2\right] = \left\{\begin{smallmatrix}\text{mod-}\mathbf{R}\\ \text{columns}\end{smallmatrix}\right\} = \Big\{ &\left[\begin{smallmatrix}0\\0\end{smallmatrix}\right], \ldots, \left[\begin{smallmatrix}23\\0\end{smallmatrix}\right], \\ &\vdots \qquad\qquad \vdots \\ &\left[\begin{smallmatrix}0\\11\end{smallmatrix}\right], \ldots, \left[\begin{smallmatrix}23\\11\end{smallmatrix}\right] \Big\} = \left\{\left[\begin{smallmatrix}m_1\\m_2\end{smallmatrix}\right] : m_1 = 0, \ldots, 23 \text{ and } m_2 = 0, \ldots, 11\right\}\end{aligned}$$

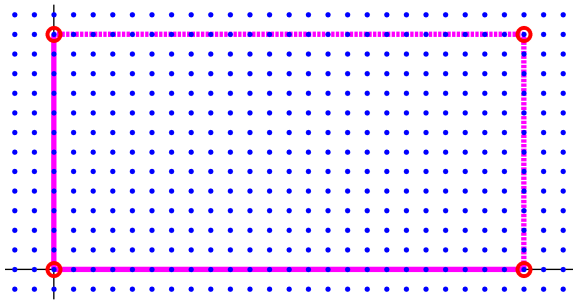
$$\begin{aligned}\left[\mathbb{Z}^2/\mathbb{Z}^2\mathbf{R}\right] = \left\{\begin{smallmatrix}\text{mod-}\mathbf{R}\\ \text{rows}\end{smallmatrix}\right\} = \Big\{ &[0\ 0], \ldots, [23\ 0], \\ &\vdots \qquad\qquad \vdots \\ &[0\ 11], \ldots, [23\ 11] \Big\} = \left\{[r_1\ r_2] : r_1 = 0, \ldots, 23 \text{ and } r_2 = 0, \ldots, 11\right\}\end{aligned}$$

and $\left\{\begin{smallmatrix}\text{mod-}\mathbf{N}\\ \text{columns}\end{smallmatrix}\right\}$ as illustrated.

The special FFT is sketched in Fig. 16, which shows $|\mathbf{L}| = 2$ FFT's of matrix size $\mathbf{R}$ followed by $|\mathbf{R}| = 24 \times 12 = 288$ FFT's of matrix size $\mathbf{L}$. The latter, by (31), have $|\mathbf{L}| = 2$ inputs and $|\mathbf{L}| = 2$ outputs, and the complex exponentials in (31) here (with $\mathbf{L}$ replacing $\mathbf{N}$ there) become ± 1 and make these matrix-size $\mathbf{L}$ DFT's into standard butterflies. The $|\mathbf{L}| = 2$ FFT's of matrix size $\mathbf{R}$ are each further decomposed, in the spirit of Fig. 14, into 24 one-dimensional 12-point column-wise FFT's followed by 12 one-dimensional 24-point row-wise FFT's.

Twiddle factors $\mathrm{e}^{-\mathrm{j}2\pi(\boldsymbol{r}\mathbf{R}^{-1})(\mathbf{L}^{-1}\boldsymbol{\ell})}$ are calculated using $\mathbf{L}^{-1} = \frac{1}{2}\left[\begin{smallmatrix}-1&-2\\-1&0\end{smallmatrix}\right]$ and $\mathbf{R}^{-1} = \frac{1}{24}\left[\begin{smallmatrix}1&0\\0&2\end{smallmatrix}\right]$ and, from them,

$$\begin{aligned}\mathbf{L}^{-1}\boldsymbol{\ell} &\in \left\{\tfrac{1}{2}\left[\begin{smallmatrix}-1&-2\\-1&0\end{smallmatrix}\right]\left[\begin{smallmatrix}0\\0\end{smallmatrix}\right], \tfrac{1}{2}\left[\begin{smallmatrix}-1&-2\\-1&0\end{smallmatrix}\right]\left[\begin{smallmatrix}-1\\0\end{smallmatrix}\right]\right\} \\ &= \left\{\left[\begin{smallmatrix}0\\0\end{smallmatrix}\right], \tfrac{1}{2}\left[\begin{smallmatrix}1\\1\end{smallmatrix}\right]\right\} \\ \boldsymbol{r}\mathbf{R}^{-1} &\in \left\{[r_1\ r_2]\tfrac{1}{24}\left[\begin{smallmatrix}1&0\\0&2\end{smallmatrix}\right] : r_1 = 0, \ldots, 23 \text{ and } r_2 = 0, \ldots, 11\right\} \\ \boldsymbol{r}\mathbf{R}^{-1}\mathbf{L}^{-1}\boldsymbol{\ell} &\in \overbrace{\{0\}}^{\text{when } \boldsymbol{\ell} = 0} \cup \overbrace{\left\{[r_1\ r_2]\tfrac{1}{48}\left[\begin{smallmatrix}1&0\\0&2\end{smallmatrix}\right]\left[\begin{smallmatrix}1\\1\end{smallmatrix}\right] : r_1 = 0, \ldots, 23 \text{ and } r_2 = 0, \ldots, 11\right\}}^{\text{when } \boldsymbol{\ell} = \left[\begin{smallmatrix}-1\\0\end{smallmatrix}\right]}\end{aligned}$$



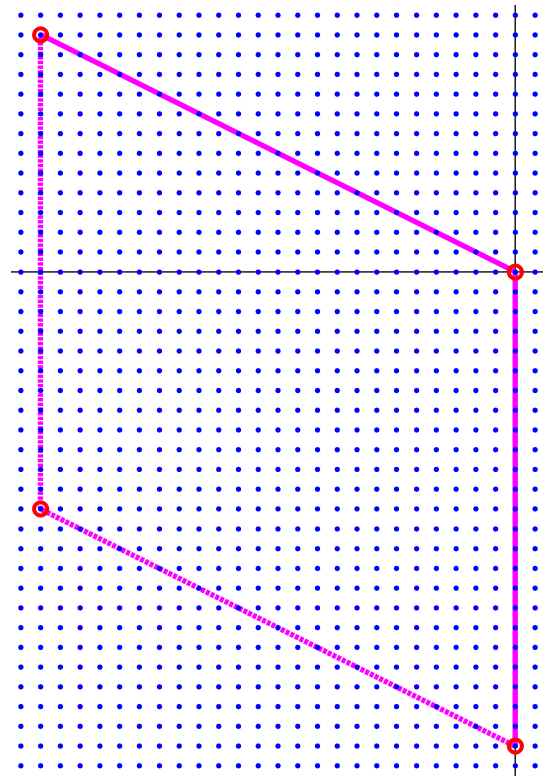


Figure 15: Above: example graphical determination of column and row remainder sets $\bmod \mathbf{R}$ using (7) and (12) respectively for $\mathbf{R} = \left[\begin{smallmatrix} 24 & 0 \\ 0 & 12 \end{smallmatrix}\right]$. Right: an example graphical determination of $\left\{\begin{smallmatrix}\text{mod-}\mathbf{N} \\ \text{columns}\end{smallmatrix}\right\}$ using (7) for $\mathbf{N} = \left[\begin{smallmatrix} 0 & -24 \\ -24 & 12 \end{smallmatrix}\right]$.

$$= \{0\} \cup \left\{ \left[\begin{smallmatrix} r_1 & r_2 \end{smallmatrix}\right] \left[\begin{smallmatrix} 1 \\ 2 \end{smallmatrix}\right] / 48 \colon r_1 = 0, \ldots, 23 \text{ and } r_2 = 0, \ldots, 11 \right\}.$$

Using (39) for the first equality, the output index

$$\begin{aligned}
\boldsymbol{k} \in \left[\mathbb{Z}^2/\mathbb{Z}^2\mathbf{N}\right] &= \left\{ \boldsymbol{s}\mathbf{R} + \boldsymbol{r} \colon \boldsymbol{s} \in \left[\mathbb{Z}^2/\mathbb{Z}^2\mathbf{L}\right], \boldsymbol{r} \in \left[\mathbb{Z}^2/\mathbb{Z}^2\mathbf{R}\right] \right\} \\
&= \left\{ \boldsymbol{s} \left[\begin{smallmatrix} 24 & 0 \\ 0 & 12 \end{smallmatrix}\right] + \boldsymbol{r} \colon \boldsymbol{s} \in \{ \left[\begin{smallmatrix} 0 & 0 \end{smallmatrix}\right], \left[\begin{smallmatrix} 0 & -1 \end{smallmatrix}\right] \}, \boldsymbol{r} \in \left\{ [r_1 \; r_2] \colon r_1 = 0, \ldots, 23 \text{ and } r_2 = 0, \ldots, 11 \right\} \right\} \\
&= \left\{ \boldsymbol{s}' + \boldsymbol{r} \colon \boldsymbol{s}' \in \{ \left[\begin{smallmatrix} 0 & 0 \end{smallmatrix}\right], \left[\begin{smallmatrix} 0 & -12 \end{smallmatrix}\right] \}, \boldsymbol{r} \in \left\{ [r_1 \; r_2] \colon r_1 = 0, \ldots, 23 \text{ and } r_2 = 0, \ldots, 11 \right\} \right\} \\
&= \left\{ \left[\begin{smallmatrix} \alpha_1 & \alpha_2 \end{smallmatrix}\right] \colon \alpha_1 = 0, \ldots, 23 \text{ and } \alpha_2 = 0, \ldots, 23 \right\}
\end{aligned}$$

steps straightforwardly through $0, \ldots, 23$ in each coordinate. The input $x_{\mathbf{L}\boldsymbol{m}+\boldsymbol{\ell}}$, however, is indexed by a vector that, due to the influence of $\mathbf{L}$ jumps through index space as the components of vector $\boldsymbol{m}$ are stepped. Thus the Fig. 16 structure can be seen to be a decimation-in-input-space structure, or, in the relevant application context here, a decimation-in-element-space structure.

5 Concluding Remarks

Steering a planar receive array in many beam directions simultaneously is most efficiently done using an array with element centers laid out on a lattice, because that enables many-beam steering using an FFT structure to process the element outputs. This paper is intended to include enough detail and example material to equip a determined system designer with the tools needed to design such a system without the artificial, unnecessary, and in the end expensive constraints often included

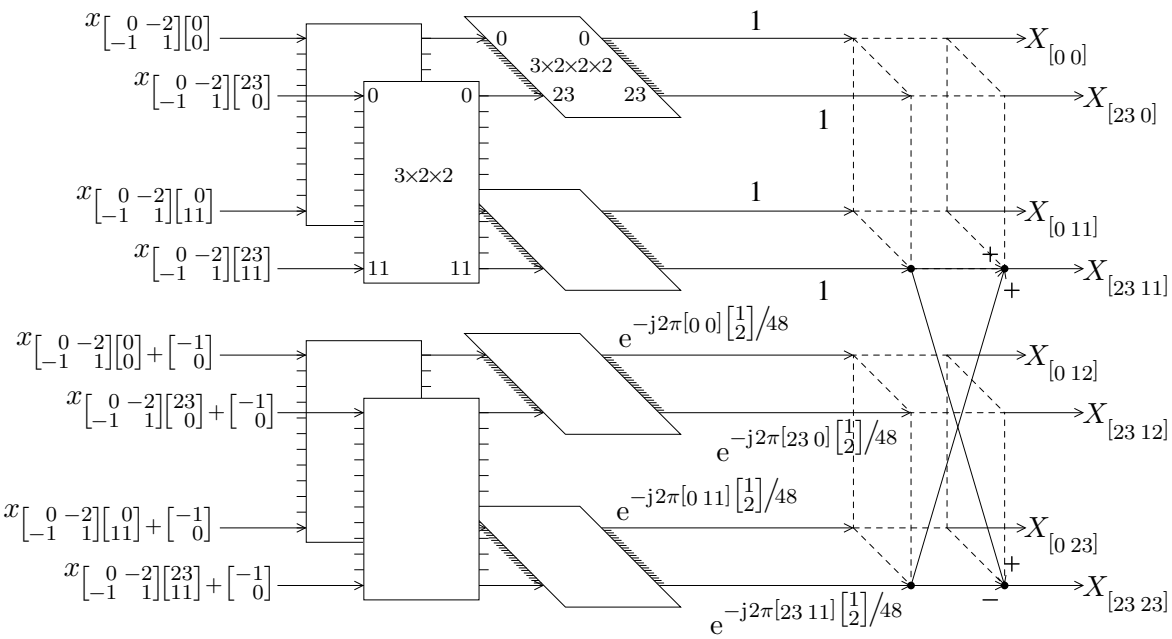


Figure 16: Example custom decimation-in-element-space beamsteering FFT for the system design example of Fig. 9. The optional $\bmod \mathbf{N}$ operation on the input subscript vector isn't shown, but if desired would use the set $\left\{\begin{smallmatrix}\text{mod-}\mathbf{N}\\ \text{columns}\end{smallmatrix}\right\}$ identified in Fig. 15. Computation is shown for all $2 \times 24 \times 12 = 576$ distinct beams, but outputs corresponding to beams not in the steering region would not be computed in practice.

---

in such designs for the sake of simplicity. This release from such constraints is of critical importance in arrays of this type, which generally require expensive digital receivers behind each element. This material in this paper is at the heart of minimizing the number of those receivers and of the amount of hardware required for subsequent beamsteering.

This paper is not intended, however, as a Treatise From God detailing step-by-step what the designer must do and making the designer more or less unnecessary in the process. Instead, considerable room is left for creative use of the body of concepts and approaches developed and illustrated here. Further, the reader is not asked to take the author's word on theoretical foundations but is instead presented with the mathematical theory in enough detail to permit verification of its correctness (or indeed to make corrections if needed). A signal-processing perspective governs throughout, rather than the more traditional electromagnetic one. Antenna-community readers might find this an irritating inconvenience, but it serves the honorable purpose of greatly simplifying the development. As unsimple as the paper surely does seem in the end, it would have been far worse if attempted without the basis matrices, remainder sets, 4D Fourier transforms, etc. used so freely throughout.

Covered in detail are the intimately related topics of the specific FFT design, the steering grid of beam directions, the grating-lobe locations in beamspace, and the element-location lattice chosen. The FFT design is based on a matrix and its factorization that together correspond to a choice of a sublattice of a sublattice of a lattice, and the FFT mathematics is then based in a deep way on coset decompositions in this chain of nested sublattices. To make this development self-contained and enabling of a general design approach, the associated coset mathematics had to be included here, but in order that the reader not need a background in group theory, that material was developed here in a self-contained way and specialized to the case at hand. The cosets were not dealt with in their usual generality, but instead the entire topic became a matter of remaindered division of an integer

vector by an integer matrix. While that material will prove the most challenging part of the paper for the typical reader, considerable effort went into attempting to making the material accessible to an MSEE engineer or to an experienced BSEE engineer with a nonphobic attitude to mathematics.

It is natural to be reading this at first on a look-ahead basis, but this author does hope that some readers have sufficiently robust reserves of engineering fortitude to be tempted to go back and work through the material of this paper carefully. Only thus can the substantial benefits of the basis-matrix notational framework to careful thinking about arrays be fully appreciated. It is in part towards that end that so many array basics are developed in detail in this paper.

References

[1] P. P. Vaidyanathan, "Multirate digital filters, filter banks, polyphase networks, and applications: a tutorial," *Proceedings of the IEEE*, vol. 78, no. 1, Jan. 1990, section IX-B, http://ieeexplore.ieee.org/iel1/5/1877/00052200.pdf.

[2] P. P. Vaidyanathan, *Multirate Systems And Filter Banks*. Prentice Hall: http://www.phptr.com/, 1993.

[3] M. Skolnik, "Attributes of the ubiquitous phased array radar," in *IEEE Int'l Symp. on Phased Array Systems and Technology*, Oct. 2003, pp. 101–106.

[4] J. O. Coleman, K. R. McPhail, P. E. Cahill, and D. P. Scholnik, "Efficient Subarray Realization through Layering," in *Antenna Applications Symposium:* http://www.ecs.umass.edu/ece/allerton/., Monticello IL, USA, Sept. 21–23, 2005.

[5] I. N. Herstein, *Topics in Algebra*, 2nd ed. Wiley: http://www.wiley.com/, 1975.

[6] J. H. Conway and N. J. A. Sloane, *Sphere Packings, Lattice and Groups*, 2nd ed. Springer Verlag: http://www.springer.com/, 1988.

[7] J. O. Coleman, D. P. Scholnik, and J. J. Brandriss, "A specification language for the optimal design of exotic FIR filters with second-order cone programs," in *Proc. 36th IEEE Asilomar Conference on Signals, Systems, and Computers:* http://www.asilomarssc.org/, Nov. 2002.